Cross-associating unlabelled timbre distributions to create expressive musical mappings

Dan Stowell DAN.STOWELL@EECS.QMUL.AC.UK
Mark D. Plumbley MARK.PLUMBLEY@EECS.QMUL.AC.UK
Centre for Digital Music, Queen Mary University of London

Editors: Tom Diethe, Nello Cristianini, John Shawe-Taylor

Abstract

In *timbre remapping* applications such as concatenative synthesis, an audio signal is used as a template, and a mapping process derives control data for some audio synthesis algorithm such that it produces a new audio signal approximating the perceived trajectory of the original sound. Timbre is a multidimensional attribute with interactions between dimensions, and the control and synthesised signals typically represent sounds with different timbral ranges, so it is non-trivial to design a search process which makes best use of the timbral variety available in the synthesiser. We first discuss our preliminary work applying standard machine-learning techniques for this purpose (PCA, self-organising maps), and the reasons they were not satisfactory. We then describe a novel regression-tree technique which learns associations between unlabelled multidimensional timbre distributions.

1. Introduction

In music, *timbre* refers to perceptual characteristics of sound other than pitch, loudness and duration (Lakatos, 2000). Timbre can in part be related to acoustic features measured from the signal, although its description (e.g. as a vector space of some fixed dimension) is not entirely clear (ibid.).

Timbre-related features can be used for musical applications such as *concatenative synthesis* (Sturm, 2006) in which criteria such as pitch/duration/timbre are used to generate new sounds by selecting and concatenating brief sounds from a database. In such applications, timbre features are measured on short overlapping regions of audio ($\sim$ 20 ms), yielding a vector time series with a sample rate of around 100 Hz. The challenge is to play back a sequence of audio samples from the database to create an audio output that is musically analogous to the input sound. Our research concerns *timbre remapping*, a generic term for any process which analyses the timbral "trajectory" of one signal and uses it to create an analogous timbral trajectory in some other audio output process. Concatenative synthesis is one example application; more generally we aim to be able to control an arbitrary sound synthesiser using a live input signal such as a voice.

One issue we encounter is that different types of sound create different distributions in our feature space. In general there are broad structural analogies between timbre distributions, but differences which complicate the mapping from one to another (see e.g. Figure 1). We wish to perform a mapping from one distribution into another, accounting for the differences in the distributions so as to produce useful musical analogies. The task is further

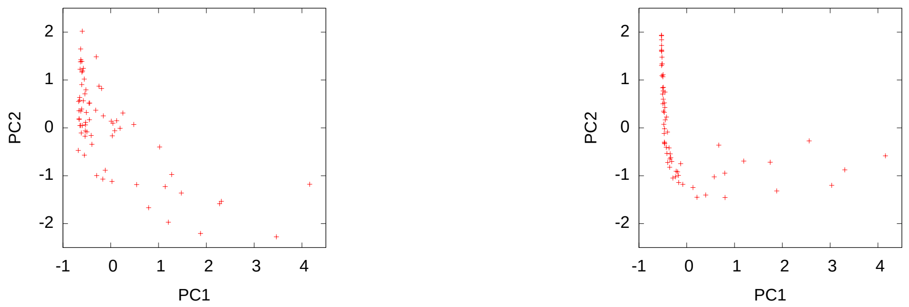


Figure 1: 2D PCA projections of timbre analysis of two sound excerpts: a drum loop (left) and thunder (right). Each point represents a 100 ms segment. The projection was calculated by applying PCA to the balanced concatenation of the separately-standardised datasets (Equation (2)).

complicated by the fact that our timbre data is unlabelled, since it is rarely feasible to annotate a large database of brief audio clips with human timbre judgements.

In this paper we first discuss two approaches based on standard machine learning techniques, which proved unsatisfactory; we then describe a variant of regression-tree learning which we designed for our purpose, and which produced improved mappings.

2. Nearest-neighbour method with PCA

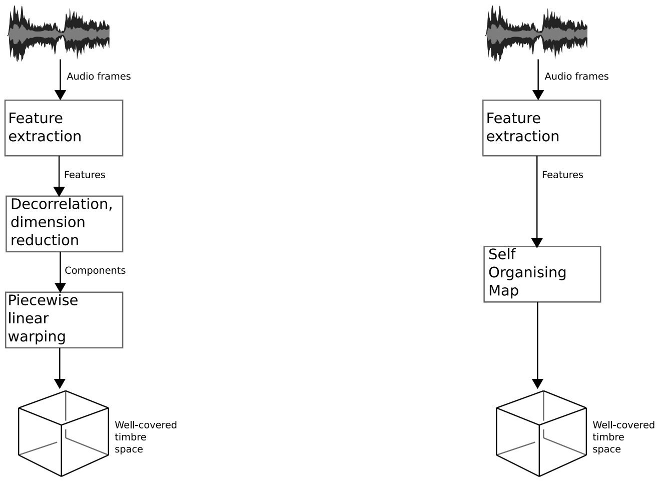
Our first approach to retrieve control parameters for each timbre co-ordinate was to use a nearest-neighbour (NN) search using Euclidean distance, as in some existing concatenative synthesis systems. To alleviate issues of the high dimensionality of the search space and of the differing distributions of timbre data, we modified this basic technique in two ways:

Dimension reduction: We applied principal components analysis (PCA) to a large set of voice timbre data, to derive a fixed projection down to only a few dimensions (typically four) which was then applied to all timbre data.

Warping: We designed a linear piecewise warping scheme (using the mean, variance, minimum and maximum of the input data), applied separately to each axis and designed to create a “well-covered” timbre space in an efficient manner for real-time use.

This approach and the piecewise warping scheme are described further in Stowell and Plumbley (2007); the flow of information processing is illustrated in Figure 2(a).

Timbre remapping in such a space is implemented by mapping an input point into the space (with a warping dependent on the source type, e.g. voice) and then performing a NN search for a datum from a training set for the target synth sound. The control settings associated with the selected NN are then sent to the synthesiser.



(a) PCA-based (Section 2) (b) SOM-based (Section 3)

Figure 2: Early approaches used to create a “well-covered” timbre space from audio data.

2.1. Issues

This PCA-based method forms the basis of a system which was used for a number of successful performances and a user evaluation study (Stowell et al., 2009). However, the approach was deemed less than perfect. Firstly the piecewise warping is a rather arbitrary approach to standardising the shapes of distributions, and has some practical problems (such as a tendency to cause rather abrupt changes in density at the boundaries of the piecewise regions). More fundamentally, the scheme is unable to account for dependences between the data axes. Since the warping is applied independently for each of the axes, it can only affect aspects of the marginal distribution, and cannot accommodate interactions in the joint distribution, such as shown in Figure 1.

The scheme could perhaps be modified to account for such issues. However, rather than pursuing that avenue we next investigated a machine-learning algorithm that seemed to offer the promise of circumventing these issues, and replacing the decorrelation and warping steps with a single coherent step: the Self-Organising Map.

3. Self-Organising Maps (SOM)

The Self-Organising Map (SOM) (Kohonen, 2001) is a simple neural-network algorithm with the potential to model the structure in our data. It consists of a single layer of “nodes” each storing a co-ordinate in the data space, and having topological connections to other nodes to create e.g. a square grid network. The training process tends to arrange the network of nodes in the data space such that the network follows the shape of a manifold implied by the training data. It can learn nonlinear manifolds and interactions between dimensions.

Important in the use of SOMs is the choice of network topology. The dimensionality of the network typically reflects the dimensionality of the manifold one hopes to recover and is typically quite low, e.g. a 1- 2- or 3-dimensional square grid of nodes. The SOM tends to adapt to the data even if the dimensionality is mismatched, but the resulting mappings may be less useful since they may contain arbitrary "twisting" of the map to fit the data. Note also that standard SOM algorithms are agnostic about the orientation of the map in the input space, meaning that any given mapping will typically take one arbitrary orientation out of many possible.

3.1. Remapping using SOMs

To prepare a SOM-based remapping system, we select a network topology and dimensionality, and then train one SOM on the timbre data for each sound source. Figure 2(b) shows the SOM-based generation of the timbre space, illustrating that the SOM replaces both the dimension reduction and the nonlinear warping of the previous PCA-based approach (Figure 2(a)).

To actually perform the timbre remapping, we map an input timbre datum onto its co-ordinate in the relevant SOM. We then retrieve the synth controls associated with the analogous position in the synth timbre data, i.e. the node at the same network co-ordinate but in the SOM trained on the synth timbre data.

This process assumes a common orientation of the SOM grids, so that a co-ordinate in one can be unambiguously held to correspond to the same co-ordinate in the other. As discussed, though, standard SOM algorithms do not guarantee this. To try and encourage a common alignment of SOM grids, we initialised the node locations as a grid oriented along the leading axes of the PCA basis used in Section 2, and we also reduced the amount by which the SOM nodes move towards training data points at each step in the training phase.

3.2. Issues

The SOM-based timbre remapping never yielded satisfactory results during development. Timbral input "gestures" tended to produce a rather arbitrary timbral output, even when the task was simplified to a very basic sound and a 2D map in a 2D feature space. From inspecting maps produced, we found that the main cause of this unsatisfactory performance was the tendency for maps to rotate and to develop twists/folds during training. This could cause undesirable mappings such as an increase in the "brightness" of the input causing a decrease in the "brightness" of the output. Despite our attempts to encourage alignment of SOM grids, there was no general setting which produced consistently useful mappings.

Twists/folds in SOMs can be caused by a poor fit of the network topology to the data. This can be considered a limitation of the standard SOM algorithm, requiring a user-specified and usually regular topology to be specified in advance. There exist related algorithms which can learn an arbitrary network topology, such as "growing neural gas" (Martinetz et al., 1993). However, applying such schemes to our timbre remapping task presents a major issue: if the map topology is learnt or adapted for each dataset, how can we map from one to another given that there will typically be no inherent one-to-one correspondence between nodes in different maps?

Such a SOM-like algorithm with adaptive topology could be the subject of future work in timbre remapping techniques. Instead we decided to approach the task of learning the structure of timbre data using a regression-tree method, which we describe next.

4. Cross-associative regression trees

A regression tree (Breiman et al., 1984, Chapter 8) is a computationally efficient nonparametric way to analyse structure in a multivariate dataset, with a continuous-valued response variable to be predicted by a set of independent variables. The core concept is to recursively partition the dataset, at each step splitting it into two subsets using a threshold on one of the independent variables (i.e. a splitting hyperplane orthogonal to one axis). The choice of split at each step is made to minimise an "impurity" criterion for the response variable summed over the subsets, often based on the mean squared error (Breiman et al., 1984, Section 8.3). The original formulation of regression trees was concerned with predicting a single univariate response variable. They were subsequently extended to multivariate responses, for example by Questier et al. (2005) who measure impurity by:

$$\text{impurity}(\alpha) = \sum_{i=1}^{n_\alpha} \sum_{j=1}^{p} (y_{ij} - \bar{y}_j)^2 \tag{1}$$

where n_α is the number of data points in the subset α under consideration, and $\bar{y}$ the sample mean of the p-dimensional response variable y_i for the points in α.

This extension yields a framework that can learn to infer relationships between one multivariate data distribution (the independent variables) and another (the response) – hence their potential application to our task. One limitation of this is that the regression is still a supervised technique, meaning that the pairwise association between items in the training datasets would need to be provided. In applications such as ours, where we might have a large database of short audio fragments from various sources, it will often be impractical to annotate the data, so next we consider an unsupervised variant.

4.1. Auto-association and multivariate splits

Questier et al. (2005) apply regression trees to the task of discovering structure in unlabelled multivariate data, by equating the response variables with the independent variables, to create an *auto-associative* multivariate regression tree (AAMRT). In other words they apply a standard regression tree with the multivariate-response extension, but there is no separation between the variables used to split the dataset and the variables whose impurity is to be minimised – the independent variables are made to "predict themselves".

There are in fact two types of multivariate extension to the standard regression tree. We have described the *multivariate-response* extension; also the choice of splitting plane can be generalised to take any orientation in the feature space, rather than using only one axis. This *multivariate-splits* extension can reduce bias in the resulting estimator (Gama, 2004). We therefore developed a regression tree based on AAMRT but multivariate in both senses.

Note that the impurity measure (1) is equivalent to the sum of variances in the subsets, up to a multiplication factor which we can disregard for the purposes of minimisation. By the law of total variance, minimising the total variance within the subsets is the same as maximising the variance of the centroids; therefore the impurity criterion selects the split which gives the largest difference of the centroids of the response variable in the subsets. If only univariate splits are allowed then this can be optimised as given by Breiman et al. (1984,

Algorithm 1: The cross-associative algorithm, XAMRT(X, Y). X and Y are the two sets of vectors between which associations will be inferred.

```
$C_X \leftarrow$ centroid of X
$C_Y \leftarrow$ centroid of Y
$J \leftarrow$ result of equation (2)
$p \leftarrow$ principal component of $J$
$X_l \leftarrow X \cap ((X - C_X) \cdot p > 0)$
$X_r \leftarrow X \cap ((X - C_X) \cdot p \leq 0)$
$Y_l \leftarrow Y \cap ((Y - C_Y) \cdot p > 0)$
$Y_r \leftarrow Y \cap ((Y - C_Y) \cdot p \leq 0)$
if $X_l$ is singular or $Y_l$ is singular then
    $L = [X_l, Y_l]$
else
    $L = \text{XAMRT}(X_l, Y_l)$
end
if $X_r$ is singular or $Y_r$ is singular then
    $R = [X_r, Y_r]$
else
    $R = \text{XAMRT}(X_r, Y_r)$
end
return [L, R]
```

Chapter 8). In the multivariate-splits variant, maximising the variance of the centroids is achieved simply by selecting the hyperplane perpendicular to the first principal component in the (centred) data. This allows for efficient implementation since the leading principal component in a dataset can be calculated efficiently e.g. by expectation-maximisation.

4.2. Cross-association

We wish to generalise the AAMRT method to apply it to two datasets defined on the same space. A simple approach would be to combine the two datasets into one and then apply AAMRT, but this would not allow the algorithm to adapt separately to the two datasets, to account for differences in location.

Instead, we modify the algorithm so that at each step of the recursion the data coming from the two distributions are *separately* centred, before a *common* principal component is calculated from their union. The recursion therefore generates two "similar but different" trees, implementing the notion that the two datasets have similarities in structure (the orientations of the splitting planes are the same) but may have differences in location at various scales (the centroids of large or small subsets of the data are allowed to differ).

To find a common principal component while giving equal priority to the two datasets (which may contain different numbers of points), we actually calculate the principal component of the concatenation J of weighted datasets:

$$J = \Big[N_Y (X - C_X) \, , \, N_X (Y - C_Y) \Big] \tag{2}$$

where X and Y represent the data (sub)sets, C_X and C_Y their centroids, and N_X and N_Y the number of points they contain.

The resulting cross-associative multivariate regression tree (XAMRT) algorithm is summarised in Algorithm 1. Note that we do not prune the tree, since for the timbral application presented here, all of the variation in the training set is useful for resynthesis.

4.3. Timbre remapping with XAMRT

To perform a remapping using a XAMRT data structure, one takes a data point and descends the tree, at each split centring it by subtracting C_X or C_Y as appropriate and then deciding which side of the splitting plane it falls. When the leaf node is reached, it contains two sets of training data points (a subset each of X and Y). If the datasets are similar in size then the leaf will often contain just one datum from each of the two distributions, giving a single choice for remapped point; if there are multiple candidates returned then a choice may be made based on other criteria (e.g. pitch matching).

We incorporated the XAMRT technique into a simple concatenative synthesis framework, and performed experiments using a range of sound recordings of different types (e.g. drum loops, thunder, beatboxing). These gave good audio results as well as making better use of the source material than a comparable NN search: the information efficiency (a normalised entropy measure) of the distribution of source material usage was improved from 70.8% (±4.4%) to 84.5% (±4.8%). For full details of the experiment see Stowell and Plumbley (2010); audio examples are available online.[1]

5. Conclusions and further work

We have described how our investigations into the mapping of musical timbre from one source onto another led to creating an unsupervised variant of multivariate tree regression. Experimental evidence indicates that it works well for our purpose. The algorithm is a batch method; as future work we will explore adapting it to enable online learning.

References

L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone. *Classification and Regression Trees.* Wadsworth Statistics/Probability Series. Wadsworth Inc, 1984.

J. Gama. Functional trees. *Machine Learning*, 55(3):219–250, 2004. doi: 10.1023/B:MACH.0000027782.67192.13.

T. Kohonen. *Self-Organizing Maps.* Springer, Berlin, 3rd edition, 2001.

S. Lakatos. A common perceptual space for harmonic and percussive timbres. *Perception & Psychophysics*, 62(7):1426–1439, 2000.

T. M. Martinetz, S. G. Berkovich, and K. J. Schulten. 'Neural-gas' network for vector quantization and its application to time-series prediction. *IEEE Transactions on Neural Networks*, 4(4):558–569, 1993. doi: 10.1109/72.238311.

[1] `http://archive.org/details/xamrtconcat2010`

F. Questier, R. Put, D. Coomans, B. Walczak, and Y. Vander Heyden. The use of CART and multivariate regression trees for supervised and unsupervised feature selection. *Chemometrics and Intelligent Laboratory Systems*, 76(1):45–54, 2005. doi: 10.1016/j.chemolab.2004.09.003.

D. Stowell and M. D. Plumbley. Pitch-aware real-time timbral remapping. In *Proceedings of the Digital Music Research Network (DMRN) Summer Conference*, Jul 2007.

D. Stowell and M. D. Plumbley. Timbre remapping through a regression-tree technique. In *Proceedings of Sound and Music Computing*, 2010.

D. Stowell, A. Robertson, N. Bryan-Kinns, and M. D. Plumbley. Evaluation of live human-computer music-making: quantitative and qualitative approaches. *International Journal of Human-Computer Studies*, 67(11):960–975, Nov 2009. doi: 10.1016/j.ijhcs.2009.05.007.

B. L. Sturm. Adaptive concatenative sound synthesis and its application to micromontage composition. *Computer Music Journal*, 30(4):46, 2006. doi: 10.1162/comj.2006.30.4.46.