CHAPTER IV

TIMBRE SIMILARITY

Timbre has proven to be a difficult attribute to define in acoustic perception, and there is little concensus as result in its underpinnings or the efforts to model it computationally. Psychoacoustics has long sought to better understand the space of timbre using subjective pairwise ratings between acoustic stimuli, but this information is costly to obtain and the generalizability of conclusions ultimately dependent on the palette of sounds considered. This chapter explores an objective, data-driven approach to the development of relative timbre spaces as a scalable alternative to this line of research. Here, instrument taxonomies are used to as a proxy for timbre similarity, and a deep convolutional network is used to project time-frequency representations of audio into a low-dimensional, semantically organized space. The quality of the resulting embeddings is demonstrated through a series of experiments, indicating that this approach shows significant promise for organizing large collections of audio samples by timbre.

1 Context

Despite its common usage in the various forms of music for centuries, a satisfactory definition of *timbre* remains elusive to this day; in fact, the one adopted by the American National Standards Institute embodies this challenge, arriving at a concept through the exclusion of others (?, ?):

> Timbre is that attribute of auditory sensation in terms of which a subject can judge that two sounds similarly presented and having the same loudness and pitch are dissimilar.

As evidenced by this definition, the very notion of "timbre" is still an open research topic in psychoacoustics. This reality is captured quite succinctly by Phillipe Manoury, who offered the following insight ():

> One of the most striking paradoxes concerning timbre is that when we knew less about it, it didn't pose much of a problem.

There are many advantages to developing a deeper understanding of timbre, from both an artistic and scientific perspective. Of particular interest to this work, however, the absence of a constructive definition —timbre is a result of X, Y, and Z— makes it difficult to directy build computational systems to characterize and compare timbres. Thus, before proceeding, it is valuable to review what is known of timbre, and prior efforts to transfer this knowledge into engineering systems.

1.1 Psychoacoustics

The perception of timbre falls under the umbrella of *psychoacoustics*, a topic of study that sits at the boundary between acoustics and psychology. Some of the earliest research in psychoacoustics was pioneered by von Helmholtz in his inquiries into the sensations of pitch and loudness (?, ?). Inquiries specific to timbre would not come until much later, due to two difficulties in experimental design. One, whereas pitch and loudness are predominantly one dimensional, it is unclear from personal introspection what the salient dimensions of timbre

might be. A subject might describe a sound as being "brighter" than another, but signal analysis is a critical tool in beginning to determine why. Additionally, researchers were limited by the kinds of stimuli they could create and use in perceptual experimentation, and thus were constrained in the space of possible parameters to explore.

With the advent of computers and continued scientific advances through the 20th century, these issues could be addressed directly, and several researchers set out to identify the existence of fundamental dimensions. This work, perfomed by Plomp (?, ?) Grey and Wessel (?, ?), among others, adopted a similar experimental design. Human subjects are presented pairs of sound stimuli and asked to rate the similarity between the two. Having collected an exhaustive set of pairwise ratings from a number of participants, multidimensional scaling is then used to project the stimuli into a low-dimesional space such that the reported relationships between these datapoints are minimally distorted; an example space is shown in Figure 6. Using this similarity model, the researcher then considers a wide array of time-frequency signal statistics, or *features*, in order to identify those that best correlate with the different dimensions. This approach has produced a useful, albeit large, set of features on which computational models have been constructed. Among the earliest were those of log-attack time, spectral centroid, and spectral spread, and were echoed later by other researchers, as in the work of Krumhansl ().

More recently, however, some have begun to recognize a few shortcomings of this approach to timbre research (?, ?). First, a timbre space derived from the multidimensional scaling of pairwise ratings is limited to the sonic palette used to produce it, and the inclusion of additional stimuli is likely to rearrange how the space is organized. For instance, the MDS model for

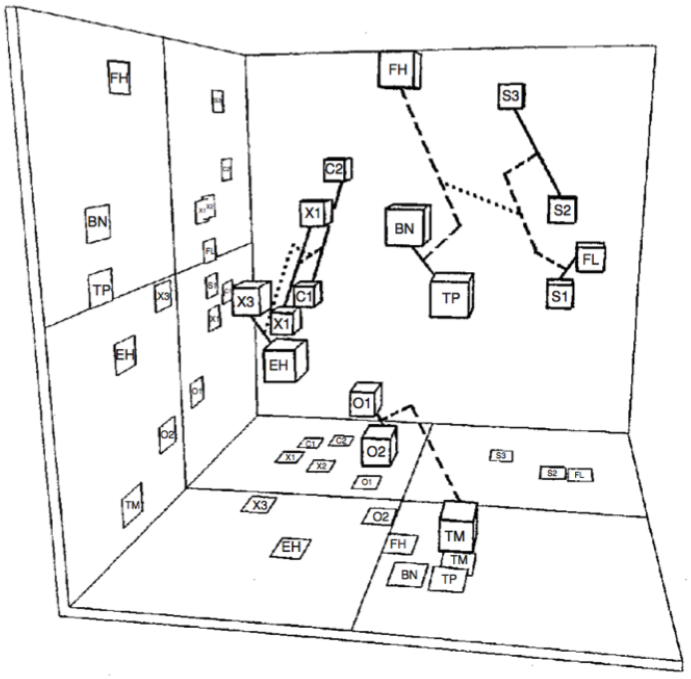

Figure 6: The resulting MDS model developed in the work of Grey and Wessel.

a collection of orchestral instruments will be quite different with and without considering electronic synthesizers. This also has significant implications on the granularity of sounds considered. In (?, ?), the attack and sustained regions of a sound were considered separately, resulting in slightly different MDS models. Additionally, concatenating the attack of one instrument with the sustained portion of another would cause a subject to perceive only the attack instrument. Second, the process of finding well-correlated features to explain the resulting MDS model is difficult and time consuming. A researcher must repeat the involved process of feature exploration for every model obtained through a different combination of stimuli. Furthermore, as noted by Caclin et al., "Given the multiplicity of acoustical parameters that could be proposed to explain perceptual dimensions, one can never be sure that the selected parameters do not merely covary with the true underlying parameters."

(?, ?). In other words, correlation does not imply causation, and features identified by inspection entail some degree of uncertainty. Finally, the process of collecting subjective pairwise ratings is especially costly, because the number of possible comparisons increases quadratically with number of unique stimuli considered. This places a practical constraint on the generality of a timbre space, as it quickly becomes impossible for subjects to exhaustively rate all combinations.

1.2 Computational Modeling of Timbre

Most previous approaches to computationally modeling timbre instantaneously can be grouped into one of two categories: signal statistics and basis projections. The first follows from the perceptual research described above, whereby specific features are designed to encode some high level semantic concept, e.g. log-attack time or spectral brightness. Initially these corresponded to the features named by in the work of Grey or Krumhansl, but have expanded over time to include a wide array of creative and clever measures. The interested reader is directed to (?, ?) for a comprehensive space of possible features.

From an often complimentary perspective, other music researchers have utilised transform-based approaches to project signals into representations with various desirable properties. One of the earliest and most common approaches is the use of Mel-frequency Cepstral Coefficients (MFCCs) for timbre-oriented tasks. Originally designed for speech coding by Mermelstein et al in the 1960s (?, ?), the first significant contribution in MIR to call attention to MFCCs as useful music features was that of Logan in 2000 (?, ?). MFCCs have, at least in practice, become nearly synonymous with timbre-centric MIR, now being used in a wide array of systems for instrument classification (?,

?), tagging (?, ?), genre prediction (?, ?), mood estimation (?, ?) or structural analysis (?, ?), to name only a few representative works in each. As described in detail in Chapter **??**, the general process of computing MFFCs proceeds as follows: an input audio signal is divided into overlapping, short-time *frames*, on the order of tens to hundreds of milliseconds; a filterbank, perceptually scaled in frequency, is then applied to each short-time frame and log-compressed; finally, a discrete cosine transform (DCT) is applied to these frequency coefficients, characterizing the shape of the spectrum (or the spectrum of the spectrum, referred to as the *ceps*trum). Often only the first dozen or so coefficients are used in practice on the principle that they capture the most relevant information, though this is more convention than rule. Some have even gone so far as to literally *equate* MFCCs and timbre, concluding that specific coefficients are responsible for various perceptual dimensions (?, ?).

Similar in principle, though less widely adopted, is to instead *learn* the set of bases against which a time-frequency representation is projected. One such instance is observed in the work Jehan (?, ?), which preserves the first 12 coefficients of a trained PCA decomposition. In this scenario, the projection into the PCA subspace serves to decorrelate the principal axes of the data in the input space, much like the Discrete Cosine Transform. The primary difference here, however, is that the bases are learned from a sample of observations, rather than defined analytically.

1.3 Motivation

While many computational approaches have proven useful for various classification or recognition tasks, none directly result in a notion of timbre similarity,

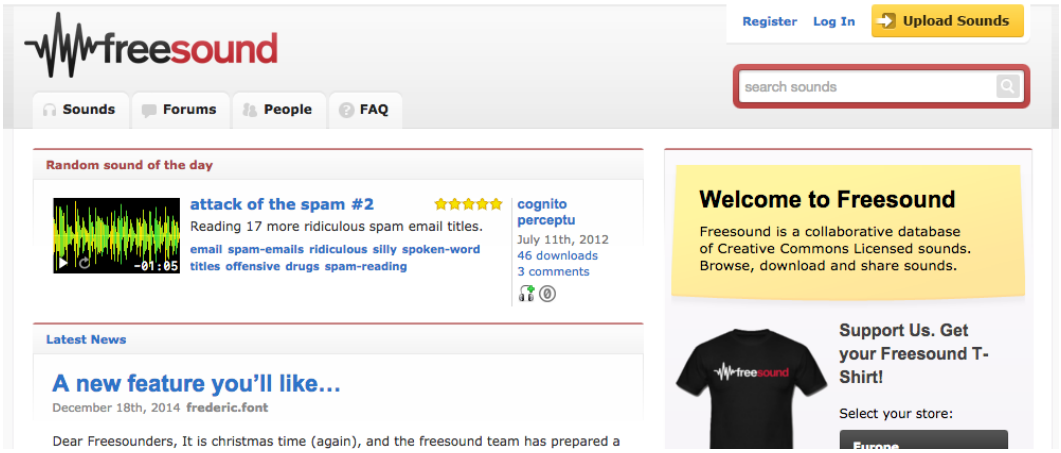

Figure 7: Screenshot of the Freesound.org homepage. Immediately visible are both the semantic descriptors ascribed to a particular sound (left), and the primary search mechanism, a text field (right).

a useful concept with a variety of applications. One notable instance is the difficulty faced in the search and navigation of large sound sample libraries. Queries are predominantly forced to take the form of text, as in the Freesound archive shown in Figure 7, which is problematic for at least two reasons. On one hand, it can challenging to describe a specific query semantically, and often metaphors and figurative language are used to relate the experience of a sound; a distorted guitar might be referred to as 'crunchy', or a trumpet as 'bright.' Conversely, this kind of descriptive language is far from standardized and varies in meaning from one individual to the next. Furthermore, such descriptions are not always associated with every sound in a collection, and typically only at the granularity of the entire recording. As a result, the task of navigating a sound library is ofetn reduced to that of an exhaustive, brute force search.

The development of a robust timbre space would not only make it possible to search for sounds with sounds, bypassing the linguistic intermediary,

but also facilitate the ranking of potentially relevant results by providing a notion of distance. This concept of a metric timbre space is also particularly attractive in the realm of user interfaces and visualization. Euclidean distance is an intuitve interaction paradigm, and visualization would allow for acoustic information to be understood in an alternative representation. The ability to explore familiar ideas from an unfamiliar perspecive holds considerable merit for artistic exploration and new approaches to composition.

1.4 Limitations

It is valuable to note that despite the difficulty inherent to defining timbre, all computational research must adopt some working concept of it, implicitly or otherwise. The work presented here operates on the assumption that the perception of timbre is tightly coupled with the experience of discriminating between unique sound sources. This is not intended to be a true equivalence with timbre, but a functional approximation that allows the research to proceed.

2 Learning Timbre Similarity

From the previous review of psychoacoustics research and efforts to computationally explain timbre, there is an important series of observations to consider. Classic timbre features are manually crafted through an involved process of inspection and exploration. When discovered, the knowledge gleaned is truly only valid in the context of the sound sources considered. As a result, the process should really be replicated for different sonic palettes, which is far from scalable. Furthermore, the subjective data necessary to conduct this kind of

research are costly to obtain. Synthesizing with the discussion from Chapter **??**, this argument makes a strong case for feature learning in timbre similarity tasks.

Having discussed the value and applications of computational timbre similarity space, it is worthwhile to outline the goals for such a system. First and foremost, one would learn, rather than design, signal-level features relevant to achieve the given task and circumvent the issues identified previously. This idea is based on the combination of an inability to clearly define the sensory phenomenon, while affording the flexibility to change the space of timbres considered. Additionally, sound should be represented in an intuitive manner, such that distance between points is semantically meaningful. In other words, signals from the same source should be near-neighbors, whereas sounds from different sources should be far apart. Finally, the ideal similarity space is perceptually *smooth*, meaning that a point that interpolates the path between two others should be a blend of the two, e.g. a tenor saxophone might fall between a clarient and a French horn.

These objectives share conceptual overlap with dimensionality reduction methods and instrument classification systems, on which this work builds. In lieu of precise information regarding the relationship between two given sounds, music instrument classes are used as a proxy for timbre similarity. The approach presented here consists of four components, as diagrammed in Figure 8, and discussed in the following subsections. First, all audio is transformed into a time-frequency representation (Subsection 2.1). The main component of the system is a deep convolutional network, which maps tiles of these time-frequency coefficients into a low-dimensional space (Subsection 2.2). A pairwise training harness is made by copying this network, and parameters

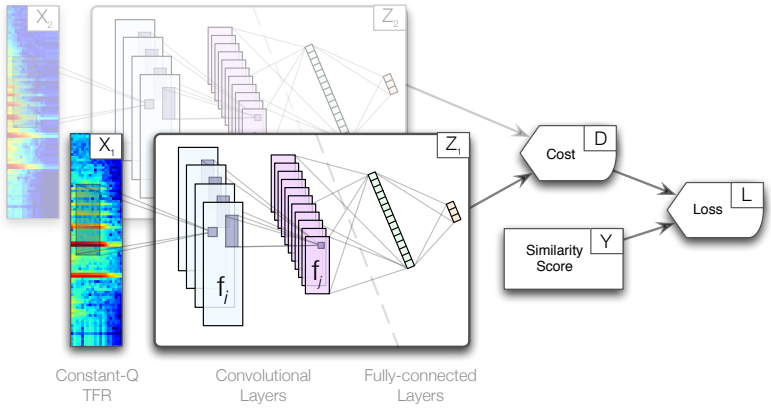

Figure 8: Diagram of the proposed system: a flexible neural network is trained in a pairwise manner to minimize the distance between similar inputs, and the inverse of dissimilar ones.

are learned by minimizing the distance between observations of the same sound source and maximizing the distance otherwise (Subsection 2.3). At test time, the pairwise harness is discarded, and the resulting network is used to project inputs to the learned embedding space.

2.1 Time-Frequency Representation

Though it is a particular goal of the system to minimally design transformations, audio is first processed by a Constant-Q transform (CQT) for three reasons. First, the application of a filterbank front-end results in a considerable simplification of the system, both computationally and in the number of learned parameters. Sharing a common formulation with neural networks, a filterbank can be viewed as a hard-coded layer in the network. Knowing the parameters in advance allows for the development of an optimized implementation, such as the one discussed in Chapter **??**, reducing processing time. Additionally, the CQT is logarithmic in frequency, serving as a reasonable

approximation of the human auditory system. Furthermore, it is generally agreed upon that timbre perception is, at least to some degree, invariant to pitch. Also following from this earlier discussion, the use of convolutional networks allows for translation invariance of features in both log_2-frequency and time.

The constant-Q filterbank is parameterized as follows: all input audio is first downsampled to 16kHz; bins are spaced at 24 per octave, or quartertone resolution, and span eight octaves, from 27.5Hz to 7040Hz; analysis is performed at a framerate of 20Hz uniformly across all frequency bins. Logarithmic compression is applied to the frequency coefficients with an offset of one, e.g. $log_{1p}(x) = log(x + 1.0)$.

2.2 Deep Convolutional Networks for Timbre Embedding

Noting that the details of deep learning and convolutional networks are discussed at length previously, only those decisions unique to this task are addressed here; for clarity regarding the mathematical or conceptual definitions of these terms, refer to Chapter **??**.

A five-layer neural network is designed to project time-frequency inputs into a low-dimensional embedding. The first three layers make use of 3D-convolutions, to take advantage of translation invariance, reduce the overall parameter space, and act as a constraint on the learning problem. Max-pooling is applied in time and frequency, to further accelerate computation by reducing the size of feature maps, and allowing a small degree of scale invariance in both directions. The final two layers are fully-connected affine transformations, the latter of which yields the embedding space. The first four hidden layers use a

hyperbolic tangent as the activation function, while the visible output layer is linear, i.e. it has no activation function in the conventional sense.

Hyperbolic tangents are chosen as the activation function for the hidden layers purely as a function of numerical stability. It was empirically observed that randomly initialized networks designed with rectified linear units instead were near impossible to train; it is hypothesized that, due to the relative nature of the learning problem, i.e. the network must discover an equilibrium for the training data, it is easy for the parameters to be pushed into a space where all activations go to zero, collapsing the network. Conversely, hyperbolic tangents are everywhere-differentiable, and did not suffer the same behavior. The use of activation functions that provide an error signal everywhere, such as sigmoids or "leaky" rectified linear units (?, ?), or better parameter initialization might avoid this behavior, but neither were explored here.

Combining the successful application of linear "bottleneck" layers (?, ?) with lessons learned from previous efforts (?, ?), the visible layer is chosen here to be linear. As will be discussed in more detail shortly, a saturating nonlinearity at the output makes the choice of hyperparameters crucial in order to prevent the network from pushing datapoints against the limits of its space. However, the absence of boundaries allows the network to find the appropriate scale factor for the embedding.

Specifically, the network is parameterized thusly: the input to the network is a 2D tile of log-CQT coefficients with shape $(20, 192)$, corresponding to time and frequency respectively; the first convolutional layer uses 20 filters with shape $(1, 5, 13)$ and max-pooling with shape $(2, 2)$; the second convolutional layer uses 40 filters with shape $(20, 5, 11)$ and max-pooling with shape $(2, 2)$; the third convolutional layer uses 80 filters with shape $(1, 1, 9)$ and

max-pooling with shape $(2, 2)$; the fourth layer is fully-connected and has 256 outputs; the final layer is also fully connected, and has 3 outputs.

2.3 Pairwise Training

As discussed previously, there are currently no known quantities with which to measure timbre, in the same way that fundamental frequency has Hertz or loudness decibels. In the absence of this absolute reference, previous efforts have instead tried to determine the relative relationships between a collection of observations. Collecting this data subjectively quickly becomes prohibitive, as the number of pairwise comparisons to be made increases quadradically with the total number of observations considered. Music instruments, however, provide an interesting source of objective information for this problem. Based on the coarse approximation that all sounds produced by a single instrument are in some sense similar, regardless of pitch or loudness, class boundaries can be used to define a neighborhood of similar timbres.

This approach to defining timbre "neighborhoods" can be used to extend the work of Hadsell et al (?, ?) to address this challenge of learning a timbre similarity space. Referred to by the authors as "dimensionality reduction by learning an invariant mapping" (DrLIM), a deep network was trained in a pairwise manner to minimize the distance between "similar" data points in a learned, nonlinear embedding space, and vice versa. Similarity was determined in an unsupervised manner by linking the k-nearest neighbors in the input space. Though left as future work, the authors propose that other information, such as class relationships, might be levereaged to learn different embeddings. This is an important consideration for the problem of timbre, be-

cause fundamental frequency and amplitude are likely to dominate the graph of nearest neighbors in the input space.

The intution behind DrLIM is both simple and satisfying: datapoints that are deemed "similar" should be close together, while those that are "dissimilar" should be far apart. Though the precise distance metric is a flexible design decision, it is used here in the Euclidean sense. A collection of similar and dissimilar relationships can be viewed as a physical system of attractive and repulsive forces. Learning proceeds by finding a balance of these contrasting forces; and furthermore, this analogy illustrates the need for both positive and negative forces to maintain equilibrium.

At its core, DrLIM is ultimately a pairwise training strategy. First, a parameterized, differentiable function, $f(\dot{|}\Theta)$, e.g. a neural network, is designed for a given problem; for the purposes of dimensionality reduction, the output will be much smaller than the input, and typically either 2 or 3 for the purposes of visualization. During training, the function f is copied and parameters, Θ, *shared* between both, such that $f_1(\dot{|}Theta) == f_2(\dot{|}Theta)$. Two inputs, X_1 and X_2, are transformed by their respective functions, f_1 and f_2, to produce the outputs, Z_1 and Z_2. A metric, e.g. Euclidean, is chosen to compute a distance, D between these outputs. Finally, a similarity score, Y, representing the relationship between X_1 and X_2, is passed to a contrastive loss function, which penalizes similar and dissimilar pairs differently. When the pair is similar, the loss will be small when the distance is small; for dissimilar pairs, the loss will be small when the distance is outside a given margin, m. This is expressed symbolically by the following:

$$Z_1 = f_1(X_1|Theta), Z_2 = f2(X_2|Theta)$$

$$D = ||Z_1 - Z_2||_2$$

$$\mathcal{L}_{sim} = D^2$$

$$\mathcal{L}_{diff} = max(0, m_{diff} - D)^2$$

$$\mathcal{L} = Y * \mathcal{L}_{sim} + (1 - Y) * \mathcal{L}_{diff}$$

Note that similarity is given by $Y = 1$, for consistency with boolean logic. As a result, the first term of the loss function is only non-zero for similar pairs, and the inverse is true for the second term.

Returning to the previous discussion regarding the dynamic range of the output layer, it should now be clear that the choice of margin only influences the learned embedding relative to a scale factor when the output is unbounded. The two loss terms are mirrored parabolas, and changing the margin, or horizontal offset, only serves to move the vertical line at which they reflect, and not the curvature of the space. In fact, this observation encourages a simple generalization of this loss function, where a second margin is introduced:

$$\mathcal{L}_{sim} = max(0, D - m_{sim})^2$$

$$\mathcal{L}_{diff} = max(0, m_{diff} - D)^2$$

$$\mathcal{L} = Y * \mathcal{L}_{sim} + (1 - Y) * \mathcal{L}_{diff}$$

Whereas the differential margin controls the variance of all points in space, the similar margin will control the variance of a similarity neighborhood. In the original formulation, where implicitly $m_{sim} = 0$, the loss is lowest when all inputs are mapped to *exactly* the same point; for the purposes of similarity, a more diffuse distribution of points is desirable. It is worth noting the slight parallel to linear discriminant analysis, which seeks to minimize intraclass variance and maximize interclass variance. Given the relative nature of this trade-off, it is sufficient to pick a single ratio between the margins, eliminating the need to vary both hyperparameters.

In practice, training proceeded via minibatch stochastic gradient descent with a constant learning rate, set at 0.02 for 25k iterations, or until a batch returned a total loss of zero. Batches consisted of 100 comparisons, drawn such that a datapoint was paired with both a positive and negative example.

3 Methodology

To assess the viability of data-driven nonlinear semantic embeddings for timbre similarity, and thus address the goals outlined at the outset of Section **??**, two experiments are used to quantify different performance criteria. First, the local structure and class boundaries of the learned embeddings are explored with a classification task. Second, global organization of the space is measured by a ranked retrieval task. Additionally, in lieu of a subjective evaluation of perceptual "smoothness" of the resulting timbre space, the learned embeddings are investigated qualitatively. In each instance, the approach presented here is compared to a conceptually similar, albeit admittedly simpler, system.

Finally, the formulation described in the previous section presents two system variables, thus giving rise to two additional considerations:

1. What is the effect of using different margin ratios?

2. How does the set of instruments considered impact the learned embedding?

3.1 Data

The data source used herein is drawn from the Vienna Symphonic Library (VSL), a truly massive collection of studio-grade orchestral instrument samples recorded over a variety of performance techniques *. In aggregate, the VSL contains over 400k sound recordings from more than 40 different instruments, both pitched and percussive. Sorting instrument classes by sample count yields 27 instruments with at least 5k samples; three of these instruments, however, are not reasonably distinct from other sources, e.g. "flute-1" and "flute-2", and discarded rather than risk introducing conflicting information. This decision yields the set of instruments contained in Table **??** for experimentation.

The distribution of sound files for these instruments, grouped by class, is given in Figure 9. As discussed previously, it is an inherent difficulty of pairwise similiarty models that the resulting relationships are limited by the number of unique classes considered. Fortunately, there is no added cost to considering a wider palette of sound sources here because the label information is objective. Therefore, building upon previous work (?, ?), three configuration subsets are repeated from the pilot study as well as a fourth consisting of all 24 classes, given in Table **??**.

* VSL Link

Table 2

Instruments considered and their corresponding codes.

Instrument	Code	Instrument	Code
French Horn	ho	Tuba	tu
Violin	vi	Cimbasso	ci
B♭ Clarinet	klb	Piccolo	pt
Tenor Trombone	tp	Oboe	ob
C Trumpet	trc	Bass Clarinet	bkl
Bass Trombone	bp	Wagner Tuba	wt
Acoustic Concert Guitar	akg	Contra Bassoon	kfa
Bassoon	fa	English Horn	eh
Cello	vc	Bass	kb
Bass Trumpet	bt	Soprano Saxophone	sxs
Distorted Guitar	eg	Tenor Saxophone	sxt
Flute	fl	Alto Flute	afl

For each instrument class, 5k samples are drawn, without replacement, to build a uniformly distributed collection. This step simplifies the process of data sampling during stochastic training of the network, which may be sensitive to class imbalances. The collection of instrument samples is stratified into five equal partitions for cross validation, used at a ratio of 3-1-1 for training, validation, and testing, respectively. The partitions are circularly rotated such that each is used as the test set once, i.e. (1, 2, 3)-4-5, (2, 3, 4)-5-1, and so on.

3.2 Margin Ratios

Though the pairwise training strategy described in Section **??** consists of two margin hyperparameters, it is ultimately the ratio between the two that gov-

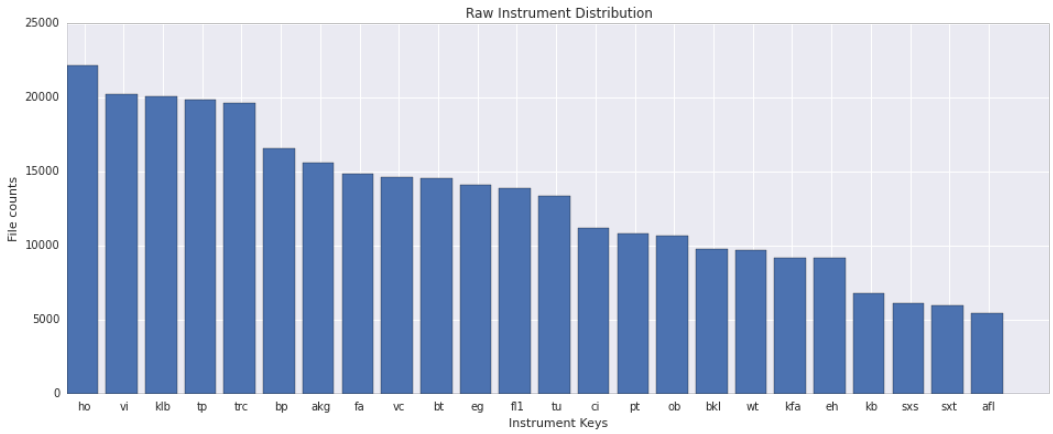

Figure 9: Distribution of instrument samples in the Vienna Symphonic Library.

Table 3

Instrument set configurations.

Key	Instrument Codes
c5	tu, ob, klb, vc, fl
c8	trc, ho, ob, eh, klb, sxt, vi, vc
c12	c8 + {tp, tu, fa, fl}
c24	c12 + {bp, akg, bt, eg, ci, pt, bkl, wt, kfa, kb, sxs, afl}

erns how the space will be shaped. In isolation, the exact choice of dissimilar term's margin, m_{diff}, is inconsequential and determines the radius of the bounding sphere. Going forward, this value is arbitrarily fixed to $\sqrt{12}$, corresponding to the radius of the sphere that intersects the coordinate $(2, 2, 2)$. Moving the similar term's margin, m_{same}, relative to this value will lead to different embeddings, and three ratios of $m_{same} : m_{diff}$ are considered here: 0, $\frac{1}{4}$, and $\frac{1}{2}$. The corresponding loss functions are shown in Figure 10.

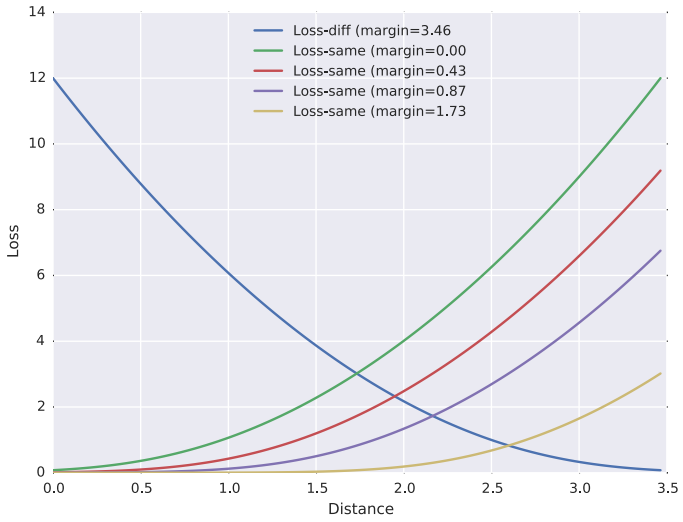

Figure 10: Distribution of instrument samples in the Vienna Symphonic Library.

3.3 Comparison Algorithm

For the purposes of comparison, a similarly motivated system is constructed using the combination of principal components analysis (PCA) and linear discriminant analysis (LDA). Previous work explored the application of PCA alone and locally linear embedding as alternative approaches to dimensionality reduction. Both are unsupervised methods, and do not make for the most fair comparison against a supervised neural network. LDA, however, is a supervised approach to dimensionality reduction, and shares at least a conceptual parallel to the proposed system, as discussed in previously in Section **??**.

It is important to note though that LDA can exhibit odd behavior in high dimensional spaces, and projecting into a PCA subspace first can help alleviate these issues (?, ?). This subspace projection is further motivated by computational efficiency concerns, where the input dimensionality is prohibitive to

training. Additionally, the cascade of PCA followed by LDA mimics a two-layer neural network, and is interchangable with the framework described here. Using the same input dimensions, 20×192), a whitened PCA transform is fit to a large sample of the training set. The principal 256 components are preserved, based on an empirical exploration of the explained variance as well as a midway point in the dimensionality reduction of the system, i.e. the number of cofficients decreases near-equally between the PCA and LDA stages. After applying the PCA transform to the training sample, an LDA transform is fit to the same data and its corresponding instrument classes, yielding a 3-dimensional embedding.

3.4 Experimental Results

As an initial quantitative inquiry, trained models are tested on a classification task using the k-Nearest Neighbors classifier in scikit-learn*. First, networks are trained across the 4 instrument configurations, 3 margin ratios, and 5 folds, and all data are projected into the resulting embedding space. From here, a collection of points are sampled from each parition –50k, 10k, 25k– for training, validation, and test, respectively. The training set is used to fit the classifier, while the validation set is used to identify an optimal setting for the parameter k, corresponding to the number of neighbors considered in the decision rule. Classification accuracy is then computed across all three sets, and tallied across folds to produce averages and standard deviations across the various conditions; these results are given in Tables **??**-**??**.

A few conclusions are immediately obvious from these results. Most

*

Table 4

k-Neighbors classification results over the training set.

config	c5	c8	c12	c24
NLSE : 0.0	93.81 ± 0.53	89.97 ± 0.40	86.70 ± 0.39	74.17 ± 0.79
NLSE : 0.25	94.21 ± 0.18	90.25 ± 0.54	87.17 ± 0.35	73.91 ± 0.61
NLSE : 0.5	93.04 ± 0.29	89.16 ± 0.27	86.08 ± 0.26	71.59 ± 0.88
PCA-LDA	64.44 ± 0.47	56.58 ± 0.60	47.22 ± 0.45	35.81 ± 0.28

Table 5

k-Neighbors classification results over the validation set.

config	c5	c8	c12	c24
NLSE : 0.0	92.37 ± 0.64	87.94 ± 0.45	84.52 ± 0.97	70.86 ± 1.51
NLSE : 0.25	93.75 ± 0.53	88.62 ± 0.58	85.65 ± 0.26	71.46 ± 0.63
NLSE : 0.5	91.75 ± 0.67	87.78 ± 0.85	83.78 ± 0.53	66.37 ± 1.69
PCA-LDA	59.97 ± 0.96	52.49 ± 2.68	39.25 ± 2.01	24.32 ± 0.97

striking is the performance discrepancy between the NLSE and the PCA-LDA models. Previous work demonstrated a significant margin between the unsupervised dimensionality reduction methods, and this result shows that the difference is indeed a function of complexity, not just the supervised learning process. To a lesser extent, all models show some degree of over-fitting, but the effect is more severe for the PCA-LDA model than any NLSE. Somewhat surprisingly, using a non-zero similiarty margin leads to slightly better classification results than the centered loss function. One explanation for such behavior is that introducing a small degree of freedom within a class may allow it to more easily form non-spherical or irregular shapes within the zero-loss volume afforded by the margin. It would appear too much freedom, on the other

Table 6

k-Neighbors classification results over the testing set.

config	c5	c8	c12	c24
NLSE : 0.0	92.49 ± 0.41	88.26 ± 0.74	84.35 ± 0.41	70.67 ± 0.55
NLSE : 0.25	92.97 ± 0.41	88.67 ± 0.79	85.16 ± 0.24	70.28 ± 0.86
NLSE : 0.5	91.96 ± 0.35	87.83 ± 0.25	84.04 ± 0.57	66.91 ± 0.57
PCA-LDA	59.91 ± 0.77	50.24 ± 1.52	39.32 ± 0.86	24.77 ± 0.51

hand, leads to fuzzy boundaries between classes and begins to compromise local structure.

The outcome of the classification experiment can also be used to inform how smooth or intuitive this space might be. To do so, confusion matrices are shown for the c12 configuration for the best NLSE, with a margin ratio of 0.25, and the PCA-LDA model, in Tables **??** and **??**, respectively.

Table 7

Confusion Matrix for c12; NLSE with a margin ratio of 0.25.

	eh	fa	fl	ho	klb	ob	sxt	tp	trc	tu	vc	vi
eh	**85.52**	1.10	0.46	3.05	1.76	3.05	0.44	0.23	2.53	0.30	0.51	1.64
fa	1.74	**85.82**	0.05	3.93	0.26	0.19	0.63	0.81	0.51	3.64	2.48	0.51
fl	0.80	0.10	**85.20**	0.90	2.19	6.00	1.67	0.14	2.33	0.07	0.33	1.17
ho	0.76	1.88	0.07	**82.52**	0.26	0.29	0.33	6.50	1.18	2.73	0.88	1.26
klb	1.23	0.40	2.46	1.16	**86.57**	3.02	1.80	0.11	1.01	0.25	1.37	1.05
ob	2.90	0.06	3.09	1.12	2.56	**81.22**	0.44	0.17	5.29	0.04	0.04	1.39
sxt	0.24	0.38	1.01	0.51	1.24	0.84	**86.34**	0.14	0.48	0.65	4.97	2.78
tp	0.39	0.87	0.10	11.87	0.03	0.49	0.20	**80.96**	2.38	2.73	0.95	0.59
trc	1.14	0.11	1.77	3.84	0.56	4.13	0.59	1.74	**83.45**	0.08	0.09	2.47
tu	0.04	1.55	0.04	5.32	0.04	0.01	0.57	2.18	0.09	**86.44**	2.82	0.57
vc	0.27	0.53	0.24	1.51	0.79	0.49	2.68	0.27	0.63	2.32	**89.74**	1.49
vi	0.49	0.23	0.61	2.44	0.46	1.20	2.46	0.48	2.05	0.41	2.06	**87.32**

Table 8

Confusion Matrix for c12; PCA-LDA.

	eh	fa	fl	ho	klb	ob	sxt	tp	trc	tu	vc	vi
eh	**46.10**	3.01	11.27	6.79	2.61	10.92	0.21	9.69	9.85	0.79	1.20	1.04
fa	5.12	**46.84**	0.37	14.42	0.31	0.27	1.88	7.55	0.58	17.15	2.47	0.57
fl	13.33	1.62	**20.82**	6.93	4.01	17.74	1.79	4.26	16.32	1.63	2.10	1.85
ho	5.45	19.49	2.11	**43.53**	1.77	0.29	0.94	15.47	1.38	7.51	1.53	4.70
klb	10.80	4.88	11.69	10.62	**9.04**	9.61	4.84	5.47	11.99	6.51	7.96	5.07
ob	15.45	0.40	17.80	1.67	3.09	**31.93**	0.56	2.42	19.92	0.69	0.81	2.14
sxt	0.48	2.77	2.28	3.62	2.33	0.97	**47.47**	1.40	3.45	9.48	14.48	15.77
tp	15.98	9.58	6.45	19.43	2.02	5.74	0.52	**18.93**	9.42	4.33	1.15	2.07
trc	10.72	0.20	14.14	3.64	3.50	19.71	0.77	5.67	**36.01**	0.27	1.12	4.97
tu	0.04	12.39	0.06	7.13	0.77	0.03	4.35	3.68	0.03	**62.74**	7.67	0.26
vc	0.39	1.65	2.87	2.66	2.09	2.36	18.99	1.16	4.29	10.02	**44.95**	12.05
vi	0.93	1.78	2.63	5.04	1.62	3.22	13.43	1.44	7.86	1.43	4.49	**56.47**

Though more confusions are to be expected in the PCA-LDA model, given the classification accuracy, it is important to note that these errors are distributed across all classes. This higher noise-floor indicates that the instruments' distribution exhibit a good deal of overlap in space. Some logical confusions seem unavoidable, such as french horn (ho) and trombone (tp), or flute (fl) and oboe (ob), occuring in both models. The former makes sense given common instrument families, i.e. brass, while the latter likely arises from the upper range of the instruments, which has fewer harmonics.

Other instrument relationships also appear to confound some element of pitch height in similarity, particularly for the PCA-LDA model. This is observed in the confusions between tuba, bassoon, and French horn. In the NLSE model, tuba is confused with French horn more often than bassoon; for the PCA-LDA model, however, the inverse is true. Intutively, the two brass instruments should share the higher confusion rate, and thus pitch is being used by the LDA model as a feature with which to distinguish between classes. The convolutional model, on the other hand, is forced to embrace a considerable amount of pitch invariance.

To help demonstrate the semantic organization of the learned embedding, 3D scatter plots are given in Figure 11 following observations of the three instruments common to all configurations –clarinet, oboe, cello– across the different embeddings for $m = 0.25$. Other instruments are displayed as semi-transparent black, to clearly highlight the three instruments of interest while giving an impression of the overall space. The main takeaways from this visualization are two-fold. First, the various sonic palettes used to learn the embedding result in different organizations of points in space. That said, the

relationship between the three sources is relatively consistent, as the cluster of clarinet always sits between oboe and cello.

To further test the semantic organziation of the learned embeddings, the outputs are used as features for a ranked retrieval task, using Euclidean distance as a scoring function. Recall-precision curves are computed using the scikit-learn toolkit and averaged over the five folds; the resulting curves for the four configurations are shown in Figure 12.

Given the classification accuracy and confusion matrices above, the discrepancies between the NLSE and PCA-LDA models is unsurprising. Still, it is interesting to consider what the shape of these recall-precision curves indicates. The two characteristics to observe are the concavity of the contour and the "knee" at which it breaks downward. In all instrument configurations, with the slight exception of "c5", the NLSE models and the PCA-LDA model exhibit opposite second-derivatives. This behavior can be understood as the acceleration with which precision changes as a function of recall. For the NLSEs, precision degrades slowly until reaching a crossover point, referred to here as the knee, where precision drops off rapidly. The PCA-LDA model does the opposite, where precision drops quickly close to a query point, and slows as recall increases. Therefore, as encouraged by the visualizations, the NLSEs contain better separated class clusters than the PCA-LDA embeddings. Furthermore, the knee of a recall-precision curves belies an interesting region in the document space, indicating that the edge of a cluster has been reached. This is a useful observation for determining early-stopping criteria in the display of ranked results, as well as identifying boundary regions in the embedding that may present interesting opportunities for sonic exploration.

4 Conclusions

In this chapter, an approach to building a computational model of timbre similarity was presented, which achieved three goals. First, the system is able to automatically learn relevant signal level features, circumventing the challenge posed by the lack of a constructive definition of timbre. Second, the resulting timbre space is semantically well-organized, as demonstrated by classification and ranked retrieval measures, and intuitive, based on a Euclidean notion of distance in a dimensionality that can be easily visualized. Lastly, the space is quantitatively smooth, such that what confusions exist correspond to instrument families or other like sounds. This was made possible by leveraging objective information about a collection of sound sources, eliminating the need for costly subjective ratings. Together, this approach to learning a timbre space shows promise for visualization and user-facing applications, such as the search and navigation of large sound libraries.

That said, there is considerable future work to be considered. All evaluation performed here is quantitative, and arguably disconnected from all subjective experience. User studies would serve to further investigate the ideas of perceptual smoothness and if or how is obtained by the learned space. Additionally, though this approach is able to make use of objective instrument taxonomies, any similiarity space obtained through pairwise comparisons is always limited by the range of inputs considered. Therefore, in order to obtain a more general timbre space, a much wider set of sound sources would need to be considered. Conversely, the intended use case of such a system may provide a constrained palette with which to operate, e.g. instrument sounds for recording engineers and environmental sounds for computational ecologists.

Finally, there are at least two other ways the sound source information could be used to train a system in a supervised manner. One, it may be advantageous to obtain subjective pairwise ratings not between all possible sounds, but rather groups or classes of sounds. These pairwise ratings could be used to train a system with soft, continuous-valued similarity ratings, rather than the binary comparison scores used here. Two, rather than defining an entire class to be similar, a hybrid approach to similiarity based on distance-based neighborhoods in the input space constrained to a single class may also lead to interesting embeddings. It is unlikely such an embedding would exhibit spherical clusters as was produced here, but points are likely to be more uniformly distributed, or diffuse, in space.

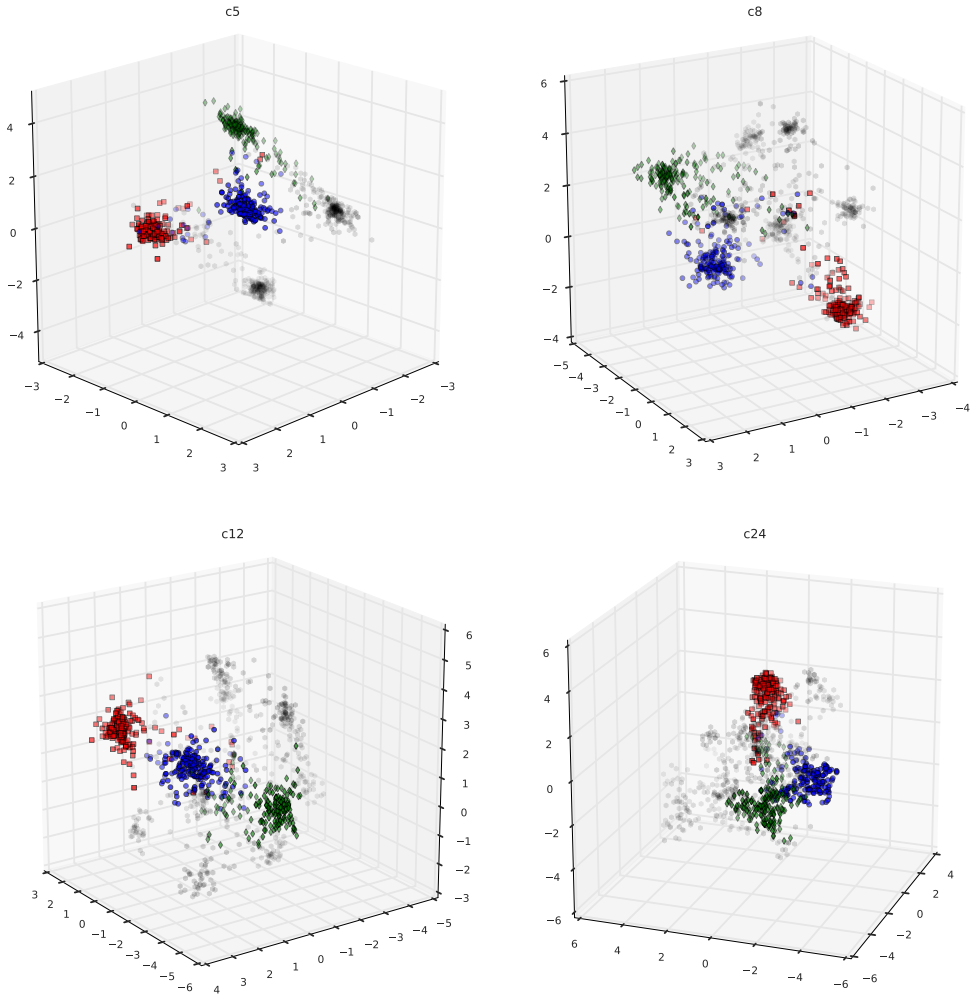

Figure 11: Embeddings of clarinet (blue circles), oboe (green diamonds), and cello (red squares) observations across models trained with the four different instrument configurations.

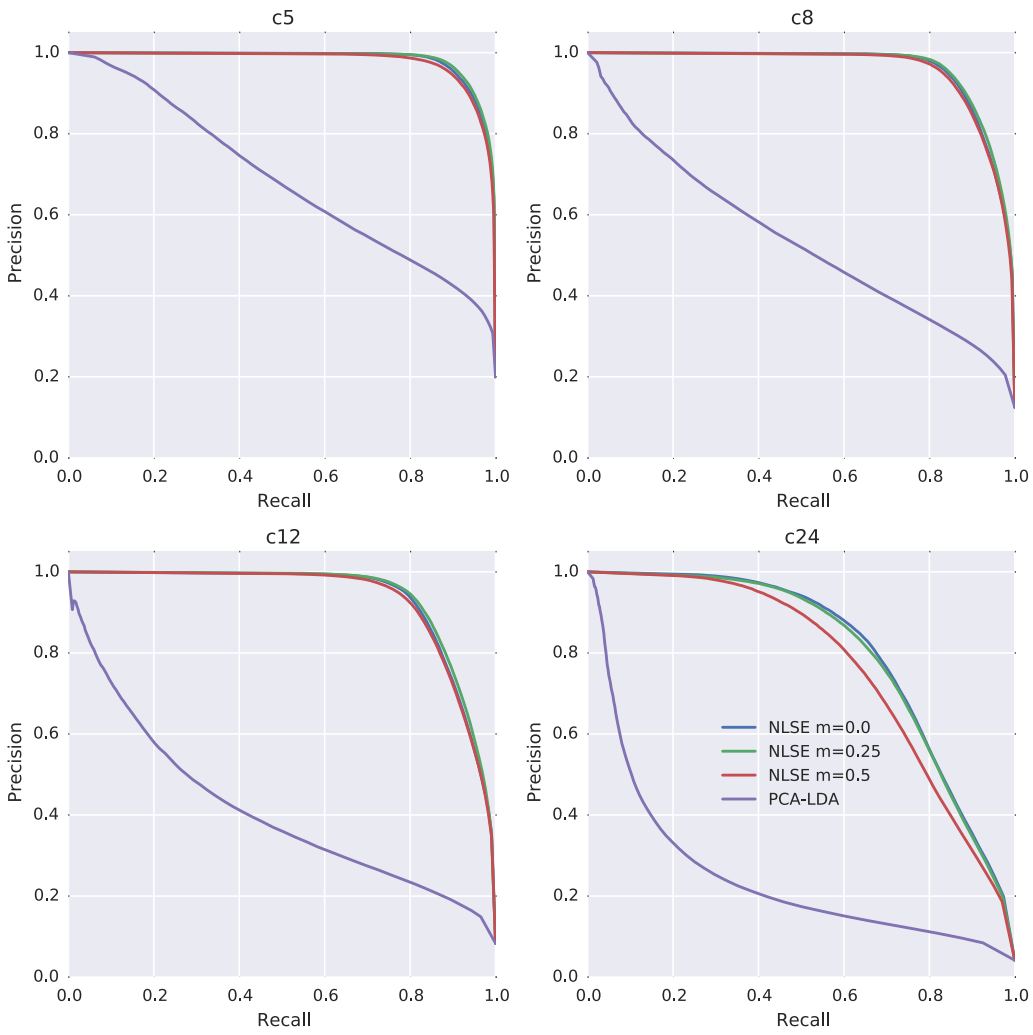

Figure 12: Recall-Precision curves over the four instrument configurations.