Research Proposal: Audio Spline Models for Machine Learning and the Role of Splines in AI and Deep Learning

Matt Klassen

DigiPen Institute of Technology, Redmond, WA 98052, USA
mattjklassen@gmail.com

Abstract. In this proposal I outline methods for modeling audio signals with splines, beginning with the simplest case of known (approximate) fundamental frequency f_0, and how these methods could be extended to more complex audio signals. The models make use of segmentation of audio signals into chunks called cycles, of approximate length $1/f_0$, which are modeled with cubic splines. Data in the model consists of a sequence of intervals which mark the cycles, and B-spline coefficients on each cycle. Interpolation is used at two levels 1) between audio samples with cubic splines, and 2) between cycles, using linear interpolation of B-spline coefficients. The latter technique is called cycle interpolation, and can lead to significant reduction of data in the model. We propose that the next step should be to design a neural network to learn the parameters of such models.

Keywords: Spline · Audio · Signal · Interpolation · Cycle · Neural · Network

1 Overview

One motivation for modeling audio signals with splines is to achieve a compressed data representation in the time domain which faithfully respresents pitch and timbre. Such models can be used to blend signals with the same pitch but different timbres, for use in sound synthesis. We showed how this can be done in [3] and [5] (also posted here: http://azrael.digipen.edu/research). To compute the models and observe properties of their graphs and audio, we work with modeling software written in C++ with JUCE. (Brief demos of this software can also be seen at the previous link.)

Another motivation for representing signals with splines is to develop low resolution models for realtime applications, such as video games. This approach is outlined in [4], where we propose a hierarchical system for audio level of detail (LOD).

From the machine learning perspective, it could be advantageous to have compressed targets for end to end generative models of audio. It may also be that the construction of spline models can be learned in an unsupervised way and the features extracted in this process could be useful for recognition and classification.

In the next two sections we outline the basic features of the audio spline models and discuss possible neural network models.

2 Audio Spline Model Features

The basic spline model for a short audio sample (a few seconds) with fundamental frequency f_0, is described in [3] as a sequence of cycles, initially determined by zero crossings, together with a spline model on each cycle. If the spline model uses roughly one third of the original audio sample data as interpolation points, the remaining sample points can be recomputed with little to no noticeable difference in audio quality. The next step is to introduce cycle interpolation, which means that a sequence of key cycles is chosen and the intermediate cycles are reconstructed from the key cycles. The simplest way to do this is to represent each key cycle with the same number n of B-spline coefficients and to use linear interpolation of B-spline coefficients for intermediate cycles. This approach relies on consistency of shape of cycles, in order to reconstruct intermediate cycles reasonably accurately. As pointed out in [3], certain types of singularities arise when using cycles based on zero crossings. To compensate for this, we developed a different approach for determining cycle endpoints, which we call the delta model.

In the delta model, described briefly in [4], cycles are not restricted to endpoints determined by zero crossings. Rather, they are determined by similarity to neighboring cycles. This is done by using the spline coefficients from the previous cycle (superimposing the previous cycle shape onto the current cycle) layered on top of a single cubic polynomial which has zero derivative at the two endpoints. This cubic 'delta' polynomial can be thought of as a slightly curved time axis over one interval representing a given cycle. The delta model allows for effective cycle interpolation with consistency between cycle shapes.

Applying these methods to signals with varying pitch and noise is challenging. The next simplest case, after instrument samples with recognizable f_0, are signals with varying f_0 separated by transition regions. If we can determine segments with consistent f_0, then it is possible to use cycle interpolation in these segments and to model each cycle independently in the transition regions.

Spline modeling of audio signals with cycle interpolation is analogous to key framing in animation. Just as key frames of an animation can be designed by artists or captured with sensors from human movement, key cycles for audio signals can be designed through methods of synthesis or can be captured with recordings. Also, intermediate frames or cycles can be generated with various interpolation schemes or algorithms. So far, we have focused on the generation of intermediate cycles using linear interpolation of B-spline coefficients and also using the capture of cycle endpoints from real audio signals.

Before describing possible neural network approaches, we specify the features of our spline representation of an audio signal, which should consist of several lists:

- list of cycle intervals (endpoints (t_i, y_i))
- list of key cycles (subset of indices of cycles)
- set of B-spline coefficients for each key cycle

We note that the cycle endpoints can be determined with continuous time t and y-values between -1 and 1 by considering a sampled audio file to be piecewise linear between samples. Also, for simplicity, we can think of the spline curve on each cycle as being represented by the same number n of B-spline coefficients, for ease of cycle interpolation, but this is not technically necessary. For instance, in order to model consecutive cycles which are particularly noisy or chaotic, one might increase the number n on some cycles.

In the next section we propose various neural network approaches to spline modeling of audio signals.

3 Proposed Neural Network Model Features

Several approaches:

Preprocessing and Audio Recognition

Constructive spline models can form a good representation for instrument samples. So it should be possible to use the essential data in such models as input to a neural network. In the case of regular cycle interpolation this can be thought of as cycle sampling. As the audio sample becomes less regular, for instance with varying f_0, the spline model requires more cycles to be key cycles. But interpolation between key cycles can still capture essential features, which may be useful for audio recognition.

Synthesis

Since the cycle interpolation model has many flexible parameters (cycle intervals, chosen key cycles, number of B-splines per cycle or cycle *dimension*, type of interpolation between key cycles), it is possible to use these for synthesis of new sounds. Although we do not have a large dataset of spline models, one could be constructed, for instance using the NSynth dataset of instrument samples (see [7]).

Unsupervised learning of spline models

This could be an interesting approach which begins with STFT (short time Fourier transform) of audio sequences to establish segments with cycle consistency based on f_0, and noisy segments as transitions between those, then doing crude classification based on key cycles from each of these segments.

Generative audio

As mentioned in the introduction, it could be advantageous to have compressed targets for end to end generative models of audio. Rather than using regression models to predict one audio sample at a time (very expensive using standard sample rates of 48000 samples per second) we propose that predicting a pattern of cycles and B-spline coefficients for selected key cycles could be sample-rate independent and computationally efficient.

Note that in the case of instrument samples, we used 18 key cycles with 33 B-spline coefficients each, so 594 or about 600 pieces of data. We could also, for a more accurate reconstruction, include the endpoints of intermediate cycles, which for $f_0 = 220$ with sample rate 44100, gives cycle length about 200 samples. So one second of audio would require $600+220 = 820$ floats, giving $820/44100 = 0.0186$, less than 2% of original data.

4 Splines and Deep Learning

I am very interested in what splines have to say about deep learning, in particular some of the foundational ideas regarding the representation of feedforward neural networks with ReLU activation function as multivariate continuous piecewise linear functions ([8], [10], [1]), and also networks with cubic spline activation ([2]). One of my goals is to do generative audio with sets of cubic B-spline coefficients as the outputs of a neural network. It may be possible to reduce the number of spline coefficients on a given cycle by using curvature methods as in [9] or to learn the placement of knots as parameters in the neural network as indicated in [10].

5 Applied Mathematics and a Conversation with chatGPT

It has fascinated me throughout my career to see how mathematics is applied in diverse contexts, such as computer science, digital signal processing, and music. One of my favorite topics, which I teach in a DSP course, is the development of Dirac's delta, first as a placeholder for a missing function attached to a Fourier series, then constructively with limits, then finally in the abstract context of tempered distributions. It is a classic example of how very useful objects can be defined by their properties in one context, then later simplified by adding a new layer of abstraction. Another example is the PLR group in transformational music theory, which I generalized to a larger group which describes the symmetries of a constraint-based system of seventh chords. From the music theory perspective, transformational approaches give simple insight into certain chord sequences, often with elegant geometric pictures and intuition. The mathematics can also take on a life of its own, leading to new questions, and sometimes leading to obscurity. The human element here is to somehow maximize relevance and beauty simultaneously.

Another example from DSP is about discrete time systems. A very well-known textbook on this subject makes a claim about LTI (linear time-invariant) systems (or filters). If the impulse response of such a system is absoltely summable, let's call the system ASIR. If the system satisfies the property that bounded input signals give bounded output signals, let's call it BIBO. The authors claim that for LTI discrete time systems, these two properties are equivalent, so "ASIR if and only if BIBO". To see that ASIR implies BIBO is straight forward. Then, to show that BIBO implies ASIR, the authors attempt to prove the contrapositive. So they assume not ASIR, then use the (not absolutely summable) impulse response to construct an output signal which blows up at time zero. They claim that this proves the result. Unfortunately, this only shows that the system has a restricted domain of signals. This brings up many interesting related questions. First, how careful should we be about the domain of a system? This is a practical matter, since we may want to use an impulse response which is not absolutely summable, and to know the behavior of the system for longer inputs. I think this was a missed opportunity which began as a desire for mathematical completeness, which in this case back-fired. This example led me to one of my first interesting discussions with chatGPT in fall of 2022. It seemed that the influence of this textbook (on the general knowledge base on the web) resulted in chatGPT supporting the "if and only if", but then backing down when asked for more evidence. In our conversation, chatGPT also offered some examples to support its claims, which led to basic errors regarding simple facts about the zeros of the sine function. This also brings up the interesting question, which I am sure will be an important topic in the current AI revolution: "How can chatGPT become better at basic math?!"

6 A Bit About Myself and my Motivation to work in AI Research

I am both a creative and analytical person. With the background and skill sets that I have, I cannot imagine a better focus than to pursue research and development in AI and its various subdisciplines. I believe that this work is one of the most critical things for humans to get right, and the time to get it right is now. I believe that AI can and will provide the means for well-informed human progress in the next few decades. AI will also be misused, but this is not a reason to step back and take a passive, or even fearful, posture regarding AI research. We don't want to find ourselves, especially those among us who are most capable of contributing to beneficial AI, in the shoes of "the best" as described by W.B. Yeats in his poem "The Second Coming":

The best lack all conviction, while the worst are full of passionate intensity. [11]

References

1. Balestriero, Randall, and Baraniuk, Richard,: *A Spline Theory of Deep Networks* Proceedings of Machine Learning Research, PMLR, (2018), 374-383.
2. Guarnieri, S., Piazza, F., Uncini, A.: *Multilayer feedforward networks with adaptive spline activation function.* IEEE Transactions on Neural Networks, 10 (3): 672-683, 1999.
3. Klassen, M.: *Spline Modeling of Audio Signals and Cycle Interpolation*, Mathematics and Computation in Music, MCM 2022, https://azrael.digipen.edu/research/
4. Klassen, M.: *Spline Modeling and Level of Detail for Audio*, Proceedings of the 19th International Conference on Signal Processing and Multimedia Applications, ISBN 978-989-758-591-3, ISSN 2184-9471, pages 94-101, https://azrael.digipen.edu/research/
5. Klassen, M., Lanthier, P.: *Design of timbre with cellular automata and B-spline interpolation*, Sound and Music Computing, (Conference) SMC 2022, https://azrael.digipen.edu/research/
6. Luebke, David, et al, *Level of Detail for 3D Graphics*, Morgan Kaufmann, Elsevier 2003, ISBN-13: 978-1-55860-838-2
7. NSynth: *The NSynth Dataset* https://magenta.tensorflow.org/datasets/nsynth
8. Parhi, R., and Nowak, R.: *What Kinds of Functions do Deep Neural Networks Learn? Insights from Variational Spline Theory.* https://arxiv.org/pdf/2105.03361.pdf, 26 Sep., 2021
9. Park, H., and Lee, J-H., *B-spline curve fitting based on adaptive curve refinement using dominant points.* Computer-Aided Design 39 (2007) 439–451.
10. Unser, Michael: *A Representer Theorem for Deep Neural Networks* Journal of Machine Learning Research, 20, (2019) 1-30.l
11. Yeats, William Butler: *The Second Coming*, https://www.poetryfoundation.org/poems/43290/the-second-coming