VAPAR SYNTH - A VARIATIONAL PARAMETRIC MODEL FOR AUDIO SYNTHESIS

Krishna Subramani , Preeti Rao

Indian Institute of Technology Bombay

subramani.krishna97@gmail.com

Alexandre D'Hooge

ENS Paris-Saclay

dhooge@crans.org

ABSTRACT

With the advent of data-driven statistical modeling and abundant computing power, researchers are turning increasingly to deep learning for audio synthesis. These methods try to model audio signals directly in the time or frequency domain. In the interest of more flexible control over the generated sound, it could be more useful to work with a parametric representation of the signal which corresponds more directly to the musical attributes such as pitch, dynamics and timbre. We present VaPar Synth - a Variational Parametric Synthesizer which utilizes a conditional variational autoencoder (CVAE) trained on a suitable parametric representation. We demonstrate[1] our proposed model's capabilities via the reconstruction and generation of instrumental tones with flexible control over their pitch.

Index Terms— Generative Models, Conditional VAE, Source-Filter Model, Spectral Modeling Synthesis

1. INTRODUCTION

Early work in audio synthesis relied on instrument and signal modeling approaches (physical and spectral modeling synthesis). Recently, there has been interesting work in the use of generative models, broadly labelled 'Neural Audio Synthesis'. These methods rely on the ability of algorithms to extract musically relevant information from vast amounts of data. Various approaches such as autoregressive modeling, Generative Adversarial Networks and VAEs have been proposed with varying degrees of success given the ultimate goal of modeling complex instrument sound sources.

Sarroff et al. [1] were among the first to use autoencoders to perform frame-wise reconstruction of short-time magnitude spectra. Roche et al. [2] extended this analysis to different autoencoder architectures, namely variational and recurrent autoencoders. They experimented with the network parameters for optimal reconstruction, and also analyzed the so called 'latent space' which is essentially a low dimensional representation of the input data. A limitation acknowledged by the above works was the lack of meaningful control over the latent space for use in synthesis. Esling et al. [3] incorporated a regularization term in the VAE latent space in order to effect some control over the perceptual timbre of synthesized instruments. With the similar aim of meaningful interpolation of timbre in audio morphing, Engel et al. [4] replaced the basic spectral autoencoder by a WaveNet [5] autoencoder.

In the present work, rather than generating new timbres, we consider the problem of synthesis of a given instrument's sound with flexible control over the pitch. Wyse [6] had the similar goal in providing additional information like pitch, velocity and instrument class to a Recurrent Neural Network to predict waveform samples more accurately. A limitation of his model was the inability to generalize to notes with pitches the network has not seen before. Défossez et al. [7] also approached the task in a similar fashion, but proposed frame-by-frame generation with LSTMs. As is well known, pitch shifting without timbre modification (i.e. preserving naturalness of the instrument sound from body resonances) requires the use of a source-filter decomposition where the filter (i.e. the spectral envelope) is kept constant during pitch transposition [8]. The other advantages of such a powerful parametric representation over raw waveform or spectrogram is the potential to achieve high quality with less training data. Recognizing this in the context of speech synthesis, Blaauw et al. [9] used a vocoder representation for speech, and then trained a VAE to model the frame-wise spectral envelope. Engel et al. [10] also recently proposed the control of a parametric model based on a deterministic autoencoder.

The VAE has the attractive properties of continuous latent variables and the additional control over the latent space by way of prior probabilities giving good reconstruction performance [11]. Our approach in this paper is to use the VAE for the modeling of the frame-wise spectral envelope similar to Blaauw et al. [9] but for instrumental sounds. Given that even for a chosen instrument, the spectral envelope is not necessarily invariant with changing pitch, we further explore the conditional VAE (CVAE), to achieve conditioning of the generation on pitch. The motivation for our work comes from the desire to synthesize realistic sounds of an instrument at pitches that may not be available in the training data. Such a context can arise in styles such as Indian art music where continuous pitch movements are integral parts of the melody. We evaluate our approach on a dataset of violin, a popular instrument in Indian music, adopted from the West, due to its human voice-like timbre and ability to produce continuous

[1] Repository : https://github.com/SubramaniKrishna/VaPar-Synth

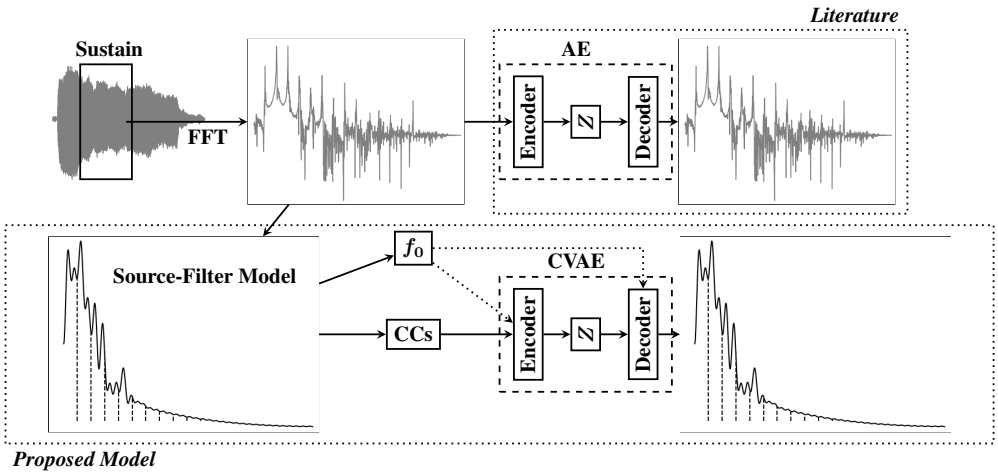


Fig. 1: Flowchart of the state of the art frame-wise audio synthesis pipeline (upper branch) and our proposed model (lower branch). Z represents the latent space learned by the (CV)AE.

pitch movements [12].

The parametric representation we adopt involves source filter decomposition applied to the harmonic component of the spectrum extracted by the harmonic model [13]. The filter is estimated as the envelope of the harmonic spectrum and represented via low-dimensional cepstral coefficients [14]. Therefore, as opposed to training a network to directly reconstruct the full magnitude spectrum as currently done in previous literature (upper branch in Figure 1), we train a CVAE on the real cepstral coefficients (CCs) conditioned on the pitch (lower branch in Figure 1). The trained network will presumably capture the implicit relationship between source and filter from the dataset samples and thus generalize better to new conditional parameter settings.

2. DATASET

We work with the good-sounds dataset [15]. It consists of two kinds of recordings (individual notes and scales) for 12 different instruments sampled at $F_s = 48\text{kHz}$. We select the violin subset of the data. The recordings have been labeled for played as good (hence good-sounds!) and bad. We use only 'good' recordings, all of which are played in mezzo-forte loudness on a single violin. We choose to work with the 4^{th} octave (MIDI 60-71) representing mid-pitch range. There are around 25 instances (recordings) per note in the selected octave. The average duration is about 4.5s per note. From each note segment, we first extract the sustained portion by applying energy thresholds. We split the data to train (80%) and test (20%) instances across MIDI note labels. We train our model with frames (duration 21.3ms) from the train instances, and evaluate model performance with frames from the test instances.

3. PROPOSED SYSTEM

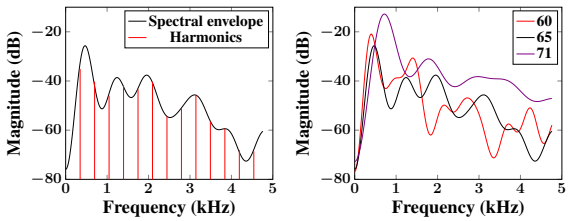
3.1. The Parametric Model

From the frame-wise magnitude spectrum, we obtain the harmonic representation using the harmonic plus residual model [13] (currently, we neglect the residual). Next, we decompose the harmonic spectrum with the source-filter model as proposed by Caetano and Rodet [16]. The filter is represented by the 'spectral envelope'. Roebel et al. [8] outline a procedure to extract the harmonic envelope using the 'True Amplitude Envelope (TAE)' algorithm (originally by Imai [17]). This method addresses the issues with the traditional cepstral liftering, where the envelope obtained tends to follow the mean energy. The TAE iteratively applies cepstral liftering to push the envelope to follow the spectral peaks. The envelope is represented by the cepstral coefficients (CCs), with the number of kept coefficients (K_{cc}) dependent on the pitch (fundamental frequency f_0) and sampling frequency as below,

$$K_{cc} \leq \frac{F_s}{2f_0}. \tag{1}$$

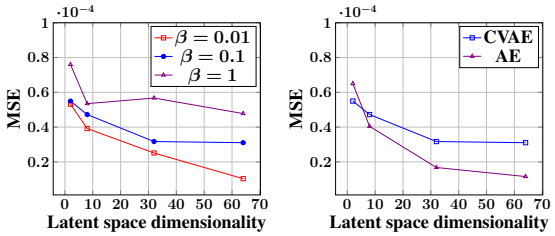
Figure 2a shows a spectral envelope extracted from one frame of a MIDI 65 instance superposed on the actual harmonics. We see that the TAE provides a smooth function from which we can accurately estimate harmonic amplitudes by sampling at the harmonic frequency locations.

The spectral envelopes for different notes appear in Figure 2b indicating clear differences in the spectral envelope shape across the range of pitches shown. This reinforces the importance of taking into account spectral envelope dependence on the pitch for musical instruments [18, 16]. It is expected that the process of estimation of the envelope itself also could contribute to the variation. Incorporating the dependency on pitch will obtain more realistic harmonic amplitude



(a) Single note (MIDI 65) (b) Different notes

Fig. 2: Spectral envelopes from the parametric model



(a) CVAE, varying β (b) CVAE($\beta = 0.1$) vs AE

Fig. 3: MSE plots to decide hyperparameters

estimates and potentially more natural synthesized sound over what is possible with a phase vocoder.

3.2. Network Architecture

We try out two kinds of Networks - Autoencoders (AE) and Conditional Variational Autoencoders (CVAE) [11, 19, 20]. AEs are trained to minimize the reconstruction loss (usually the mean squared error (MSE)) between the input and output. In our case, the network input is the cepstral coefficients, and thus MSE represents a perceptually relevant distance in terms of squared error between the input and reconstructed log magnitude spectral envelopes. VAEs enforce a Gaussian prior on the latent space by learning probabilistic encoders and decoders, and optimizing the Variational Lower Bound [11, 19] given by,

$$\mathcal{L} = \mathbb{E}_{z\sim Q}\{\log P(X|z)\} - \beta D_{KL}\{Q(z|X)||P(z)\}, \quad (2)$$

where the first term models the reconstruction loss, and the second term enforces the desired prior on the latent space. β controls the relative weighting of the two terms [21]. CVAEs learn a conditional distribution on a variable (f_0 in our case), and minimize a slightly modified VAE loss function [19, 20]. The motivation to use a CVAE is two fold:

1. It allows us to obtain a continuous latent space from which we can sample points (and synthesize the corresponding audio).

2. By conditioning on the pitch, we expect the network to capture the subtle dependencies between the timbre and the pitch, thus allowing us to generate the envelope more accurately, and at the same time giving us the ability to control the pitch.

The main hyperparameters in our networks are the dimensionality of the latent space and the value of β. To decide these, we train the network on the train data instances with different hyperparameters, and evaluate the networks MSE with the test instances. The MSE reported here is the average reconstruction error across all the test instances. Figure 3a shows the MSE for $\beta = [0.01, 0.1, 1]$. From Equation 2, we see that high β forces gaussianity at the expense of MSE, and low β forces the network to behave like an autoencoder. From our trials, we conclude $\beta = 0.1$ to be the sweet spot to tradeoff between both of these. With this value of β, Figure 3b shows the MSE plots for AE and CVAE for latent space dimensions $[2, 8, 32, 64]$. We note a steep fall until 32, becoming more flat later, indicating that 32 is a good choice for latent space dimensionality.

All networks are implemented in PyTorch [22]. For both AE and CVAE, we work with a similar network architecture - an encoder having dimensions : $[91, 91, 32]$, and a decoder having the same (but reversed) architecture. According to Equation 1, different pitches can have different number of CCs. 91 is the number of CCs for the lowest pitch (MIDI 60). For the higher pitches, the CCs are zero padded to the same dimension 91. All the layers are linear fully connected layers and use leaky ReLu activations to allow for stable training. The optimization was performed using ADAM [23] with an initial learning rate of 10^{-3}, and training was run for 2000 epochs with a batch size of 512 on an NVIDIA GeForce GTX 1070 Mobile GPU.

4. EXPERIMENTS

While we present results in this paper for the AE and CVAE on the parametric representation of the frame-wise magnitude spectrum, we have also carried out similar experiments directly with the frame-wise spectrum. As expected the reconstruction performance is relatively poor; the complete results are reported in our accompanying repository We perform the following 2 kinds of experiments to demonstrate the capabilities of our model.

1. Reconstruction - We omit all instances of certain selected pitches during training, and see how well our model can reconstruct a note of the unseen target pitch. The spectral envelope of a note instance of the target pitch is input to the network. The output of the network is the reconstructed envelope, to be evaluated with respect to the input.

2. Generation - The purely 'synthesis' aspect of our model; we see how well our model can generate note instances with new unseen pitches.

4.1. Reconstruction

We consider two distinct training contexts for the reconstruction of a note with unseen pitch. (a) all instances of the neighbouring MIDI notes upto 3 neighbours are included in the training set, as shown in Figure 4; this is performed for $T = [63, 64, 65, 66, 67, 68]$. (b) the training set contains instances of only the octave endpoint MIDI notes, 60 and 71; we reconstruct instances of all the intermediate notes.

MIDI	T - 3	T - 2	T - 1	T	T + 1	T + 2	T + 3
Kept	✓	✓	✓	×	✓	✓	✓

Fig. 4: ✓ indicate MIDI note instances included in the training set for the synthesis of a given target note of MIDI label T.

In each of the above cases, we compute the MSE as the frame-wise spectral envelope match across all frames of all the target instances. The results are presented in Figure 5. We can see that the CVAE produces better reconstruction, especially when the target pitch is far from the pitches available in the training data. In the latter case, the MSE is seen to decrease as the target pitch moves closer to its nearer octave end pitch in both networks, as one might expect. Overall, the conditioning provided by the CVAE helps to capture the pitch dependency of the spectral envelope more accurately.

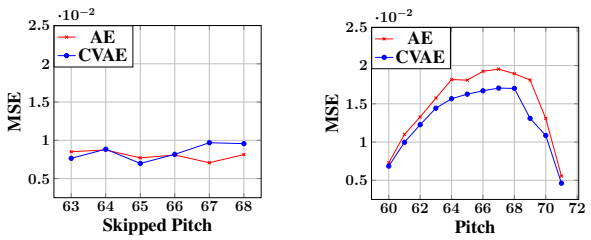


Fig. 5: Spectral envelope MSE across unseen pitch note instances with close MIDI neighbours in training data (left), and only octave end notes in training data (right).

We mention here that the same experiment with the spectral magnitude representation (rather than the parametric representation via spectral envelope) gives poor reconstructions in the form of distorted spectra that lack even a clear harmonic structure (sound examples available in our accompanying repository).

4.2. Generation

The previous experiment evaluated the networks reconstruction capabilities. However, we are ultimately interested in using it as a synthesizer. Thus, in this experiment, we see how well the network can generate the spectral envelope of an instance of a desired pitch (not available in the training data of the network). For this, we train on instances across the entire octave sans MIDI 65, and then generate MIDI 65. Generation comes naturally to the CVAE, as we just have to sample latent points from the prior distribution, and pass them through the decoder along with the conditional parameter f_0 to generate the spectral envelope (lower branch in Figure 1). Since a single latent variable represents a single frame, we have to coherently sample multiple latent variables and decode them to obtain multiple contiguous frames. Our approach to sample points from the latent space is motivated from [9] i.e. we perform a random walk with a small step size near the origin in the latent space to sample points coherently. We synthesize the audio by sampling the envelope at the harmonics of the specified f_0, and perform a sinusoidal reconstruction. We do not have an objective measure to evaluate the quality of the generated note. However informal listening indicates that it sounds close to the natural violin sound except for the missing soft noisy sound of the bowing. Incorporating residual modeling in the parametric representation in future would help restore this.

Further, we try to generate a practically useful output, viz. a vibrato violin note with typical vibrato parameters. This exercise involves reconstructing spectral envelopes corresponding to the continuum in the neighbourhood of the note MIDI pitch. In this case as well, the generated vibrato tone sounded natural in informal listening. More formal subjective listening tests are planned involving also synthesis of larger pitch movements or melodic ornaments from Indian raga music. It must be recalled that we have not taken loudness dynamics into account. All our dataset instances were labeled mezzo-forte. However a more complete system will involve capturing spectral envelope dependencies on both pitch and loudness dynamics.

5. CONCLUSION

The goal of this work was to explore autoencoder frameworks in generative models for audio synthesis of instrumental tones. We critically reviewed recent approaches and identified the problem of natural synthesis with flexible pitch control. We then presented VaPar Synth - our model to generate audio. Through our parametric representation, we can decouple the 'timbre' and 'pitch', and can thus rely on the network to model the inter-dependencies. We use a variational model as it gives us the ability to directly sample points from the latent space. Moreover, by conditioning on the pitch, we can generate the learnt spectral envelope for that pitch (something which would not be possible in a vanilla VAE), thus giving us the power to vary the pitch contour continuously in principle. We then present a few experiments demonstrating the capabilities of our model. To the best of our knowledge, we have not come across any work using a parametric model for musical tones in the neural synthesis framework, especially exploiting the conditioning function of the CVAE.

6. ACKNOWLEDGEMENTS

The authors thank Prof. Xavier Serra for insightful discussions on the problem.

7. REFERENCES

[1] Andy M Sarroff and Michael A Casey, "Musical audio synthesis using autoencoding neural nets," in *ICMC*, 2014.

[2] Fanny Roche, Thomas Hueber, Samuel Limier, and Laurent Girin, "Autoencoders for music sound modeling: a comparison of linear, shallow, deep, recurrent and variational models," *arXiv preprint arXiv:1806.04096*, 2018.

[3] Philippe Esling, Adrien Bitton, et al., "Generative timbre spaces: regularizing variational auto-encoders with perceptual metrics," *arXiv preprint arXiv:1805.08501*, 2018.

[4] Jesse Engel, Cinjon Resnick, Adam Roberts, Sander Dieleman, Mohammad Norouzi, Douglas Eck, and Karen Simonyan, "Neural audio synthesis of musical notes with wavenet autoencoders," in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, 2017, pp. 1068–1077.

[5] Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu, "Wavenet: A generative model for raw audio," *arXiv preprint arXiv:1609.03499*, 2016.

[6] Lonce Wyse, "Real-valued parametric conditioning of an rnn for interactive sound synthesis," *arXiv preprint arXiv:1805.10808*, 2018.

[7] Alexandre Défossez, Neil Zeghidour, Nicolas Usunier, Léon Bottou, and Francis Bach, "Sing: Symbol-to-instrument neural generator," in *Advances in Neural Information Processing Systems*, 2018, pp. 9041–9051.

[8] Axel Roebel and Xavier Rodet, "Efficient spectral envelope estimation and its application to pitch shifting and envelope preservation," in *International Conference on Digital Audio Effects*, Madrid, Spain, Sept. 2005, pp. 30–35, cote interne IRCAM: Roebel05b.

[9] Merlijn Blaauw and Jordi Bonada, "Modeling and transforming speech using variational autoencoders.," in *Interspeech*, 2016, pp. 1770–1774.

[10] Jesse Engel, Lamtharn Hantrakul, Chenjie Gu, and Adam Roberts, "Ddsp: Differentiable digital signal processing," *arXiv preprint arXiv:2001.04643*, 2020.

[11] Diederik P Kingma and Max Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[12] Chris Haigh, *Indian violin*, 2014 [Accessed: 21-Oct-2019], `http://www.fiddlingaround.co.uk/india/`.

[13] Xavier Serra et al., "Musical sound modeling with sinusoids plus noise," *Musical signal processing*, pp. 91–122, 1997.

[14] Marcelo Caetano and Xavier Rodet, "Musical instrument sound morphing guided by perceptually motivated features," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 8, pp. 1666–1675, 2013.

[15] Oriol Romani Picas, Hector Parra Rodriguez, Dara Dabiri, Hiroshi Tokuda, Wataru Hariya, Koji Oishi, and Xavier Serra, "A real-time system for measuring sound goodness in instrumental sounds," in *Audio Engineering Society Convention 138*. Audio Engineering Society, 2015.

[16] Marcelo Caetano and Xavier Rodet, "A source-filter model for musical instrument sound transformation," in *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2012, pp. 137–140.

[17] S. IMAI, "Spectral envelope extraction by improved cepstrum," *IEICE*, vol. 62, pp. 217–228, 1979.

[18] Wayne Slawson, "The color of sound: a theoretical study in musical timbre," *Music Theory Spectrum*, vol. 3, pp. 132–141, 1981.

[19] Carl Doersch, "Tutorial on variational autoencoders," *arXiv preprint arXiv:1606.05908*, 2016.

[20] Kihyuk Sohn, Honglak Lee, and Xinchen Yan, "Learning structured output representation using deep conditional generative models," in *Advances in neural information processing systems*, 2015, pp. 3483–3491.

[21] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner, "beta-vae: Learning basic visual concepts with a constrained variational framework.," *Iclr*, vol. 2, no. 5, pp. 6, 2017.

[22] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer, "Automatic differentiation in PyTorch," in *NIPS Autodiff Workshop*, 2017.

[23] Diederik P Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.