IMPROVING THE ACOUSTIC MODELS OF SPEECH SYNTHESIS

PRASANNA KUMAR MUTHUKUMAR

Prof. Alan W Black (chair)

Prof. Bhiksha Raj
Prof. Richard Stern
Prof. H. Timothy Bunnell

June 16, 2014 – version 0.947

Prasanna Kumar Muthukumar: *Improving the Acoustic Models of Speech Synthesis*

ABSTRACT

Statistical Parametric Speech Synthesis has been successful in producing highly understandable speech but the result is usually buzzy, robotic, and somewhat unlikeable. One major reason for this is the inadequate modeling of the human speech production system. Speech is traditionally modeled using a source-filter framework with overly simplistic assumptions of the source function. In the first part of the proposal, I describe the results obtained when more sophisticated and synthesis-appropriate models of the source function are used. I then draft a plan for future directions of investigation in this research area.

Complex source models alone do not solve the problems of synthesis though. While a variety of source-filter representations have been proposed for speech, few are suitable for use with modern statistical and machine learning techniques. One possible solution is to project these unsuitable models into a suitable space where machine learning techniques can be used. Preliminary experiments have revealed that a deep learning approach might provide the key to solving this problem. In the second part of the proposal, I explain the reasons behind this choice of technique and provide details of the experiments.

Elaborate models and complicated machine learning techniques are only useful if there exist objective metrics that can tell us how effective these two are. In the third part of the proposal, I highlight the shortcomings of current objective metrics and sketch out my ideas for an improved objective metric for Statistical Parametric Speech Synthesis.

CONTENTS

Part I

THE PROBLEMS OF PARAMETRIC SPEECH SYNTHESIS

1 INTRODUCTION

Despite the advances made in Statistical Parametric Speech Synthesis [57] in the last decade, the quality of the synthesized speech leaves a lot to be desired. Statistical Parametric Speech Synthesis assumes a parametric model of the human speech production system, and then learns a statistical mapping between the characters of text and the parameters of the model. This type of synthesis is fast, has a small memory footprint, is more robust to train-test mismatches, and requires substantially less training data. Yet, even the best of these systems do not sound as natural as Unit-Selection Synthesizers[27], which synthesize speech by concatenating *units* (typically phones, diphones, or syllables) of speech. We must therefore ask why Statistical Parametric Speech Synthesizers have difficulty synthesizing natural-sounding speech.

The first problem lies in the way that the human speech production system is parameterized. The parametric model used in most modern speech synthesis systems is the source-filter model first proposed by Fant in [15]. While many techniques have succeeded in adequately representing the filter part of the model, few have been effective in describing the source part. In Part ii of the proposal, I will summarize the various problems in modeling the source, describe my own efforts at handling this unwieldy problem, and propose future directions of exploration.

The second problem is caused by the fact that the parametric model and the machine learning techique that is used to predict the values of the parameters are developed completely independent of each other. Even though it is necessary for the parametric model and the machine learning algorithm to be deeply tied together, each of these is never explicitly designed to work well with the other. In Part iii, I will describe my attempts at bridging this gap and propose a technique that is designed to move the two closer together.

Any researcher working on either of these problems will quickly realize that making design decisions based on human evaluations is non-trivial. It is difficult for humans to choose between systems where the differences are subtle. Current objective metrics of speech quality are usually designed for speech coding tasks and do not take into account the nuances of statistical parametric speech synthesis. These metrics also tend to be orthogonal to the excitation modeling techniques and so do not measure the quality of the excitation models in any way. In Part iv, I will describe the requirements for an objective

metric for speech synthesis, and flesh out my plan for an investigation in this area.

In the last Part, I will consolidate the proposed work which was spread across all the previous parts and provide an estimated timeline for bringing this work to completion.

Part II

IMPROVED PARAMETERIZATIONS

2

BACKGROUND AND PREVIOUS WORK

The source-filter framework first proposed by Gunnar Fant has been the most successful model of the human speech production system[15]. The intuition behind this model is obvious when looking at a sagittal section of the human vocal tract shown in Figure 1.

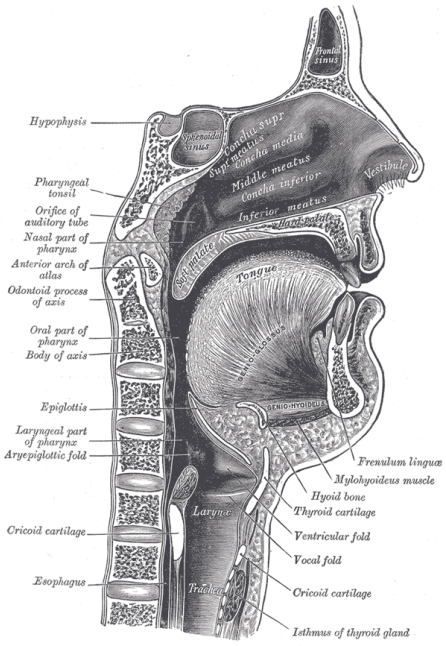
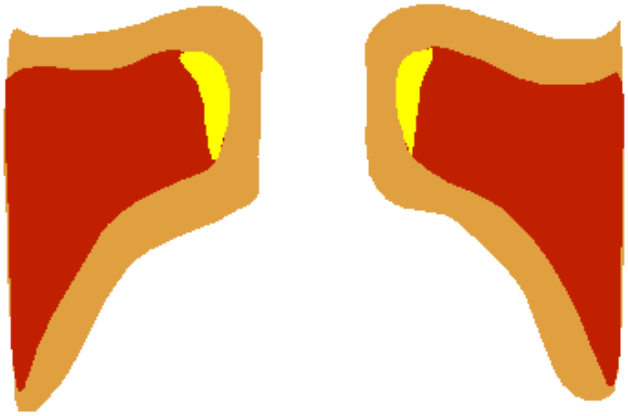


Figure 1: *Human vocal tract*

The initiation of human speech production is by the vibration of the vocal folds attached to the larynx. The sound that is produced here passes through the rest of the throat, the mouth, and the lips. This sets up resonant patterns in the vocal tract. These resonant patterns appear as formants on the spectrogram. For many speech processing applications such as speech recognition, speaker identification, etc.., it suffices to use these formants alone. Therefore, a great deal of resources have been dedicated to modeling the vocal tract resonances very accurately. Significantly less attention has been paid to modeling the vibration of the vocal folds, but the accurate modeling of these is essential for synthesis of natural sounding speech.



The animation shown above depicts the action of the vocal folds in speech when speaking voiced sounds. Most parametric models used in synthesis assume that the vocal cords only produce two kinds of sounds, a series of spikes at the fundamental frequency F0 for voiced sounds and white noise for unvoiced sounds. This assumption is quite simplistic since the vocal fold sounds actually have more structure. An additional complication is the imperfect separation of the source and the filter characteristics. Even highly successful techniques such as STRAIGHT[29] and its various flavors[30], do not completely separate the glottal source signal from the vocal tract characteristics, thereby adding another layer of complexity to the source signal.

It is therefore sensible to use excitation models with the explicit intention of modeling the characteristics of the source. These models can then represent the whole gamut of shapes that can occur in the source function both because of poor source-filter separation as well as due to the natural variation in the source function itself.

Source modeling itself is a fairly well established research area with plenty of detailed studies by Fant and others more than 20 years ago [17][16][18]. A detailed overview of all the 'classic' source models is available in [21]. In addition to these models which were created to study the source for its own sake, there have also been several models proposed purely for the sake of synthesis such as [12][49]. Despite the existence of all of these models, a quick perusal through the systems submitted to the various Blizzard challenges[1] of the last several years indicates that none of these are used when the quality of synthesis really matters. In fact, the most sophisticated source model used in these challenges is still a simple mixed excitation model[56] which merely assumes a mixture of pulse and noise rather than choosing between the two. All of these facts strongly indicate that the source modeling problem is still far from being solved.

3

AN ITERATIVE SOURCE FITTING METHOD

The models described in the previous chapter use a variety of different techniques to fit the models to the source signal. All of these techniques assume that the only available signal to do the fitting to is one isolated speech signal. This constraint is far more severe than is necessary for statistical parametric speech synthesis. When building a synthetic voice, we usually have access to a corpus of speech so we will have multiple instances of the same phone being said, and correspondingly, multiple instances of the same sort of glottal shape that our excitation model can fit to. Making full use of multiple instances of phones can greatly aid us in producing good fits to the source signal. In [36], we describe one such method to make full use of the presence of multiple instances of phones to produce better fits of the model to the source signal.

In order to fit our favorite model to the source signal, we must first extract the source from the speech signal itself. To do this, we use the Iterative Adaptive Inverse Filtering (IAIF) technique which was first described in [3]. This technique works by iteratively making estimates of the glottal source spectrum and the vocal tract spectrum, and refining estimates of one to get a better estimate of the other. The final estimate of the glottal source spectrum thus obtained is far better than could have been obtained by simple inverse filtering methods. The specific technique we used is more or less identical to the setup described comprehensively in [40]. A block diagram describing the IAIF procedure is shown in Figure 2.

The result of Iterative Adaptive Inverse Filtering is a set of LPC coefficients (which we convert to LSPs) that provide an estimate of the vocal tract spectrum and the glottal source function (the outputs of the two shaded blocks in Figure 2).

The source model that we will use to fit the residual to is the classic Liljencrants-Fant model first proposed in [19]. This model was developed as a mathematical description of the glottal flow derivative. The *derivative* has been found in practice to be easier to model compared to the glottal flow itself.

The model itself consists of the following equation:

$$e(t) = \begin{cases} E_0 e^{\alpha t} \sin(\omega_g t) & t < T_e \\ \frac{-E_0}{\varepsilon T_a} \cdot [e^{-\varepsilon(t-T_e)} - \varepsilon e^{(T_c - T_e)}] & T_e < t < T_c \end{cases} \quad (1)$$

Figure 3 shows a plot of the LF model for typical values of the parameters. The parameters T_p, T_e, T_a, T_c are explained in the figure.

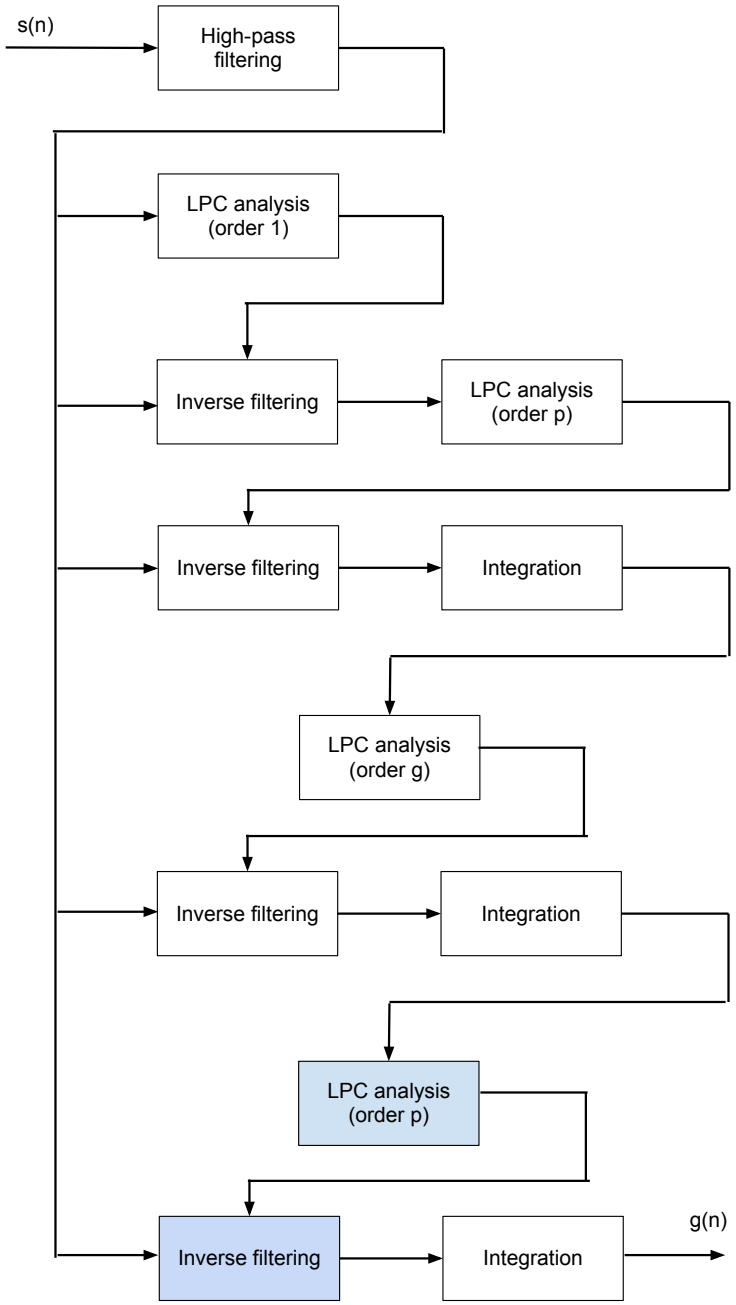


Figure 2: *Iterative Adaptive Inverse Filtering*

The parameters E_0, α and ε can be determined from the positive peak. The glottal frequency ω_g can be determined from the fundamental period T_0.

As must be obvious when looking at the plot, the LF model is intended to be fit to the source function at the peak of the glottal pulse.

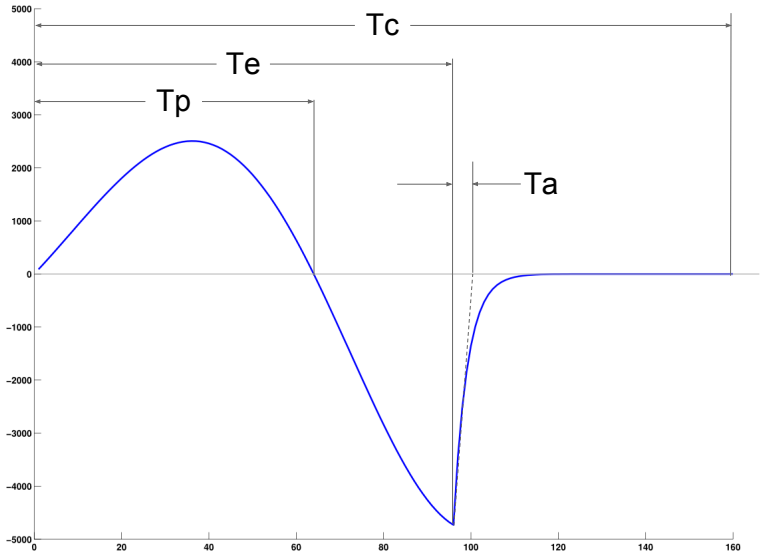


Figure 3: *LF model for typical values of the parameters*

To do this fitting, we must first detect the *location* of this peak in the source signal. We begin by extracting frames of the source signal that are at least 3 pitch periods long. Typically, we use analysis windows 70ms wide with 5ms shifts. The glottal closure instance itself is detected by convolving a *model* LF pitch pulse with this frame. The magnitudes, locations, and spacing of peaks in the convolved signal are used to estimate voicing, pitch period locations, and F_0 within the window.

If this detection process indicated that the frame was voiced, i.e. a glottal pulse was present, then the LF model is fit to the glottal pulse by a simple optimization method: the pitch period duration T_0 is held constant and the three temporal parameters, T_p, T_e, and T_a are adjusted one sample time forward or backward. Similarly, the amplitude term was adjusted by 1dB on each iteration. If the adjustement of any of these parameters decreased the RMS error between the LF model pulse and the pulse in the source signal, then the new parameter value was kept. Otherwise, the change was discarded and the direction of change of that parameter reversed in the subsequent iteration. The optimization process is stopped when no change in parameter values leads to a decrease in the RMS error.

We must remember that the LF model is only an approximation of the shapes in the glottal flow derivative. The derivative contains a lot of high frequency content that this model was not intended to capture and nor does it do so. Without these high frequency components, the synthesized speech tends to sound hollow and muffled. To model these components, we subtract the fitted LF model from the glottal

derivative. A low order LPC is then fit to the remaining components that are not captured by the LF model. This type of source model is very similar to the one proposed in [54].

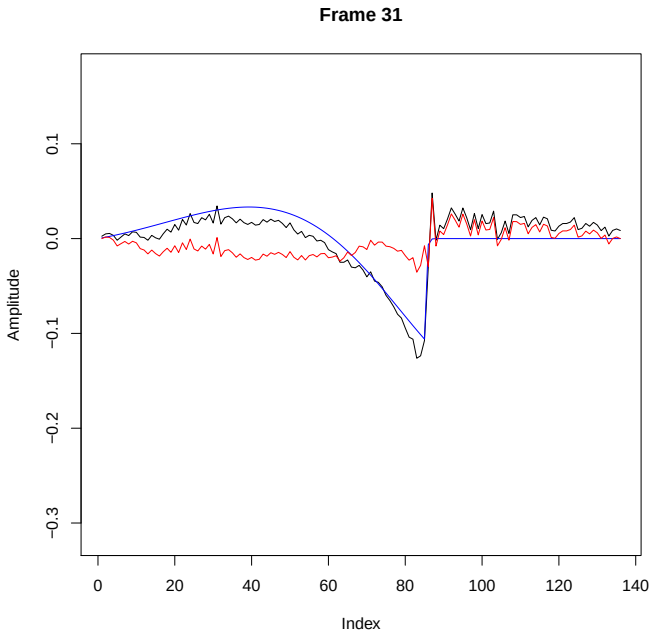


Figure 4: *Fitting to the residual: the raw glottal flow derivative is in black, the estimated LF model is in blue and the residual error is in red*

The fitting process described above is not very different from the numerous algorithms that have been proposed to do this in the past. Like all of the earlier algorithms, it still assumes that the LF model is to be fit to a *single* source signal. Whereas, for synthesis, there exist numerous examples of each phone and therefore numerous instances of the appropriate source function. Ideally, the fitting process should take full advantage of these instances to obtain a better fit than by merely fitting the model to one instance of the source signal.

One good way to take the whole corpus into account is to use the statistics of the other instances of a particular phone when fitting to the phone. We accomplish this by doing the fitting process over multiple passes of the training data. In the first iteration, the LF model is fit to each glottal pulse naively assuming that there are no other instances available. These LF parameters are then used along with the vocal tract model to build a synthetic voice in the ClusterGen framework[7]. ClusterGen builds a set of Classification And Regression Trees (CARTs)[8] that learn a mapping between the context-dependent phones and feature vectors (LF parameters + vocal tract parameters). This mapping that is learned is a sophisticated way of creating a model of the LF parameters of the source signal from the entire corpus.

In the second pass LF fitting process, the ClusterGen synthesizer considers the available lexical and phonetic information and uses the CARTs to make a prediction of the LF parameters for the glottal pulse at hand. These predicted LF parameters are then used as seed values both for the model shape that is convolved to detect the glottal pulse and as an initialization for the fitting process described earlier. Once this second pass fitting has been completed, this process of fitting and building CARTs can be repeated multiple times. By iteratively using CARTs to initialize the LF fitting and feeding the fitting result back into CART training, we end up with a very good final estimate of the true LF parameters. In addition to using information about other instances of the source signal, this iterative approach to fitting the model to the signal also has a smoothing effect which helps to remove outliers.

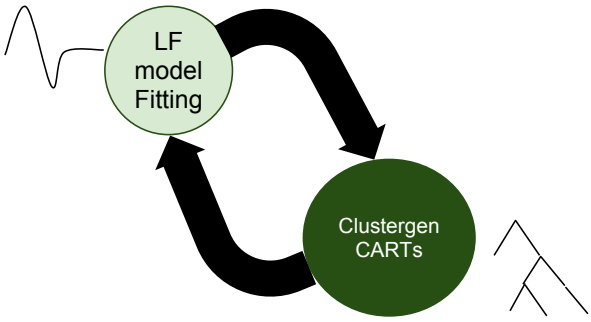


Figure 5: *Iterative estimation of LF parameters*

OBJECTIVE AND SUBJECTIVE EVALUATIONS

To test the quality of our models, we conducted listening tests on Amazon Mechanical Turk using the Testvox framework[37] where listeners were asked to choose between two systems. The first system was a synthesizer that used Line Spectral Pairs[28] and the LF model (first pass) for the residual. The second was a baseline system that used an identical vocal tract model but used a mixed excitation (ME) residual[56]. Human listeners judged the LF model synthesized speech to be more natural. Detailed statistical analyses of the listening test results are available in the paper.

We also ran listening tests that compared the results of the fitting process in the first pass to the fitting process in subsequent passes. The results however were inconclusive and the difference in quality between iterations was subtle. Even speech researchers who listened to the synthesized speech from the first and last iterations acknowledged that the speech sounded different but had difficulty making a judgement on which one sounded more natural.

We therefore had to rely on objective metrics. We were primarily interested in two specific measures. The first was the prediction error in ClusterGen. This is the error that we get as a result of the limitations of the CARTs that ClusterGen uses. The second metric was the RMSE and Correlation of the fitting process itself. These were computed for every pitch period where the LF model was being fit. While a low value for the second metric means that the fitting process works well, this is useless if the source model parameters cannot be predicted from text easily. Therefore, both the prediction error metric as well as the fitting error metric must be low for high quality synthesis to be possible. Table 1 summarizes the results of both the fitting as well as prediction errors computed on a held out test set of the same speaker. Our iterative method performs well on both metrics.

Table 1: *RMSE and Correlation of Fitting*

	LF Fitting		Prediction
Iteration number	RMSE	Corr	RMSE
0	406.89	0.482	4.840
1	405.94	0.479	6.909
5	395.17	0.518	5.611
10	391.57	0.534	5.061
15	389.98	0.543	4.836
20	389.65	0.547	4.722
25	390.06	0.550	4.661
30	390.56	0.551	4.636

All results are for the RMS voice from the CMU Arctic database[32]. Similar results were obtained for other speakers from the same corpus but will not be reported here in the interest of brevity.

4

PROPOSED WORK

Figure 6, generated by H. Timothy Bunnell, shows a segment of the residual obtained through the Iterative Adaptive Inverse Filtering process described in the previous chapter. The lower half of the plot shows the LF models after being fit to the plot in the upper half of the plot.

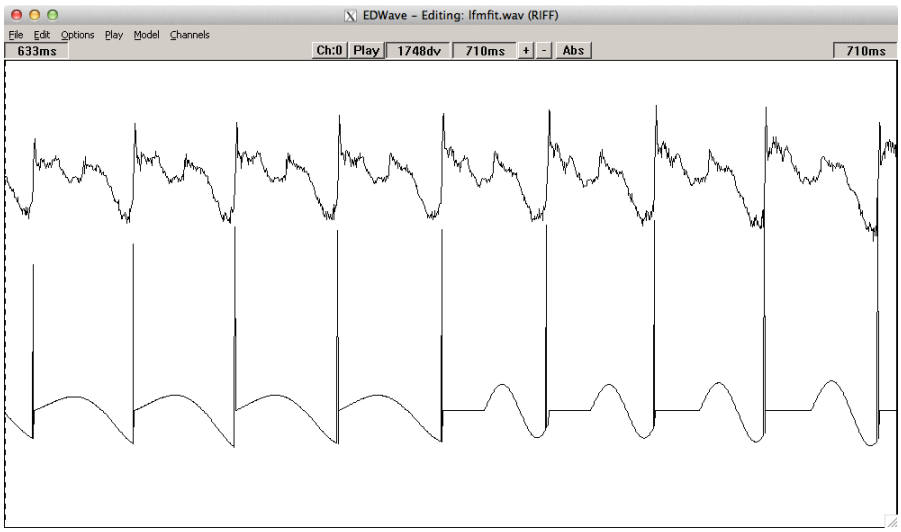


Figure 6: *Source function and fitted LF model*

As must be obvious when looking at the plot, the residual is heavily influenced by the second harmonic of the fundamental frequency. There is no way that the LF model can accurately represent these shapes. Even if the higher frequencies of the signal were removed, the residual still contains two peaks and a trough per cycle while the LF model itself only has *one* peak and a trough. This problem isn't specific to the LF model; none of the classic source models[21] are capable of handling source functions like these. This isn't actually a flaw in these source models because these models were designed to model a source signal free from the influences of the vocal tract. But, for all its merits, even the IAIF procedure cannot completely remove *all* vocal tract characteristics from the source signal.

We could try to fix this by proposing a new source model that is capable of handling shapes like the ones shown in Figure 6. However, this would only have the effect of creating yet another source model in addition to the multitude that existed at the time of Fujisaki's paper and newer ones like [9]. And this new source model would only be useful as long as the IAIF procedure is being used. Any improvements to the IAIF procedure or a switch to better vo-

cal tract representations would render our 'improved' source model useless.

I therefore propose an approach which will be a hybrid model consisting of a parametric model like the LF model combined with a non-parametric technique like Drugman's technique in [12]. The reasoning behind this is that while residuals obtained through the various vocal tract filtering techniques vary a lot, nearly all of them have an LF-like underlying structure at the glottal closure instances. The LF model could be used to model this underlying structure while whatever could not be captured by it could be represented using a non-parametric model like Drugman's Deterministic plus Stochastic model.

This proposed model is related to the one presented in the previous chapter where low-order linear predictive coding is used instead of a non-parametric model. However, in our experiments we found that LPC was insufficient to model the necessary characteristics of the source signal. It could also be argued that we could dispense with the parametric representation altogether and only use Drugman's representation. But in experiments with the ClusterGen speech synthesizer, we found that the predictability of Drugman's parameterization was low. In addition to this, parametric representations are also of a much lower dimensionality compared to non-parametric models.

Engineering arguments against Drugman's parameterization aside, we also want to try creating a new non-parametric model for another reason. We want to try to separate the components of the source signal into predictable and non-predictable components. The only thing in a speech signal that is purely unpredictable is white noise. Therefore we must create a model that tries to separate the white noise portion of the residual from the parts with more structure. It is convenient to think of this as a model in the analysis by synthesis framework where the model parameters are adjusted so that the resulting error is white noise-like.

The exact details of this model still requires a certain amount of thought and a great deal of reading. I will be able to provide a more concrete idea in another month or so.

Part III

IMPROVED MODELING

5

BACKGROUND

The speech coder used in modern Statistical Parametric Speech Synthesis [57] has remained largely unchanged for a number of years. The standard coding technique is usually a variant of Mel Cepstral analysis[52]. While many different parameterizations of the spectrum have been developed for synthesis[13][49][14][35] few have yet managed to survive in the long run. The most obvious indications of this are the systems that are submitted to the annual Blizzard Challenge[1]. Very few statistical parametric systems submitted to the challenge since its inception use vocoders that do not use Mel Cepstral coefficients. Even highly successful techniques like the various flavors of STRAIGHT[30] are rarely used by the synthesizer directly. These are usually converted into Mel Cepstral coefficients (MCEPs) before being used by statistical parametrical systems.

This lack of new parameterizations that perform better than MCEPs is especially intriguing considering the amount of research effort that has gone into finding a replacement. An ideal parameterization for statistical parametric synthesis will have to fulfill all of the following requirements:

- It must be invertible
- It must be robust to corruption by noise
- It must be of sufficiently low dimension
- It must be in an interpolable space

Even if a parameterization technique were invented that could comply with three of the above four requirements, the technique would be useless if it did not at least partially satisfy the remaining one. Therein lies the difficulty of inventing a new parameterization. Mel Cepstral coefficients satisfy all of these requirements to a reasonable extent. However, this representation is not perfect and places a major bottleneck on the naturalness of modern parametric speech synthesizers. Techniques such as [53] and [51] rectify some of the problems that occur with this representation but the Mel Cepstral representation still leaves plenty of room for improvement.

Machine learning has made a vast impact on nearly every problem in computer science today. So, rather than try to come up with a human designed parameterization for the vocoder, could we instead use machine learning algorithms to automatically create a parameterization for us that will fit all of the above requirements?

6

DEEP PARAMETERIZATIONS

Neural networks themselves have existed for many years but the training algorithms that had been used were incapable of effectively training networks that had a large number of hidden layers[34]. This is because the standard technique used for training a neural network is the backpropagation algorithm[43]. The algorithm works by propagating the errors made by the neural network at the output layer back to hidden layers and then adjusting the weights of the hidden layers using gradient descent or other techniques to minimize this error. When the network is very deep, the propagated error to the first few hidden layers becomes very small. As a result, the parameters of the first few layers change very little in training. One strategy that was developed in recent years was to start off by training the neural network one pair of layers at a time and then building the next pair on top of previous ones[25][24]. This step is called *pretraining* because the weights that are obtained through this process are used as the initialization for the backpropagation algorithm. Pretraining techniques are believed to provide an initialization much closer to the global optimum compared to the random initializations that were originally used.

Our search for a technique to create a purely data-driven parameterization led us to the Stacked Denoising Autoencoder (SDA) which was developed for pretraining deep neural networks[55]. The SDA is trained in a manner more or less identical to the layer-wise pretraining procedure described in [5] and [25]. As the name suggests, the Stacked Denoising Autoencoder is constructed by stacking several Denoising Autoencoders together to form a deep neural network. Each Denoising Autoencoder is a neural network that is trained such that it reconstructs the correct input sequence from an artificially corrupted version of the input provided to it. This process is shown in Figure 7. The network is fully connected between each layer but in the interest of clarity, the figure will only show a limited number of connections.

The SDA is of particular interest to parametric speech synthesis because this network learns to reconstruct a noisy version of the input from a lower dimensional set of features. If we use the output of one of the middle layers as a parameterization, this will by definition satisfy the first three of our four requirements. We will discuss the fourth requirement in a later section.

The SDA is actually rarely used in a task where the input needs to be *reconstructed* from the representation that the SDA transforms the

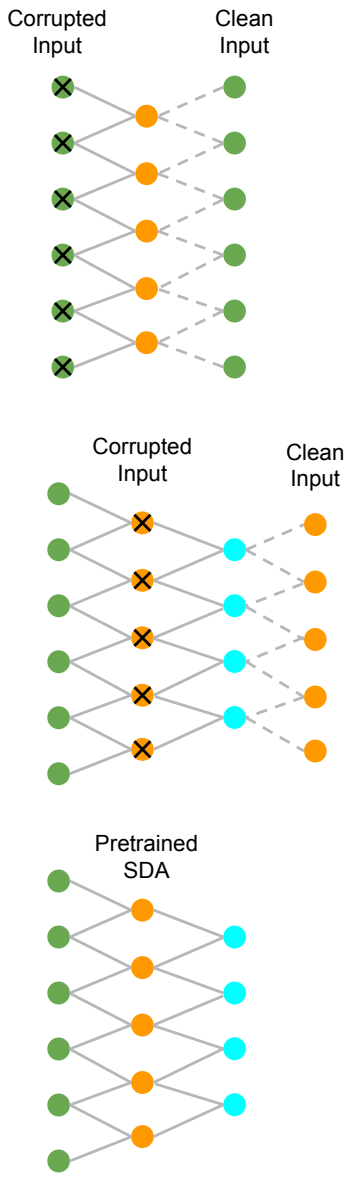


Figure 7: *SDA Pretraining*

input into. It is nearly always used to provide a lower dimensional representation on top of which a classifier such as logistic regression, or Support Vector Machines are used. An example of this is the Deep Bottleneck Features that are used in Speech Recognition[23][22]. However, such approaches are less relevant to parametric synthesis which is *not* a classification problem.

BUILDING ENCODING AND DECODING NETWORKS

The 'pretraining' process for our approach is identical to the one for speech recognition. We build an SDA on our features by stacking

multiple Denoising Autoencoders that were built by learning to reconstruct corrupted versions of the input. Once the SDA is trained, we then *unwrap* the SDA as shown in Figure 8.

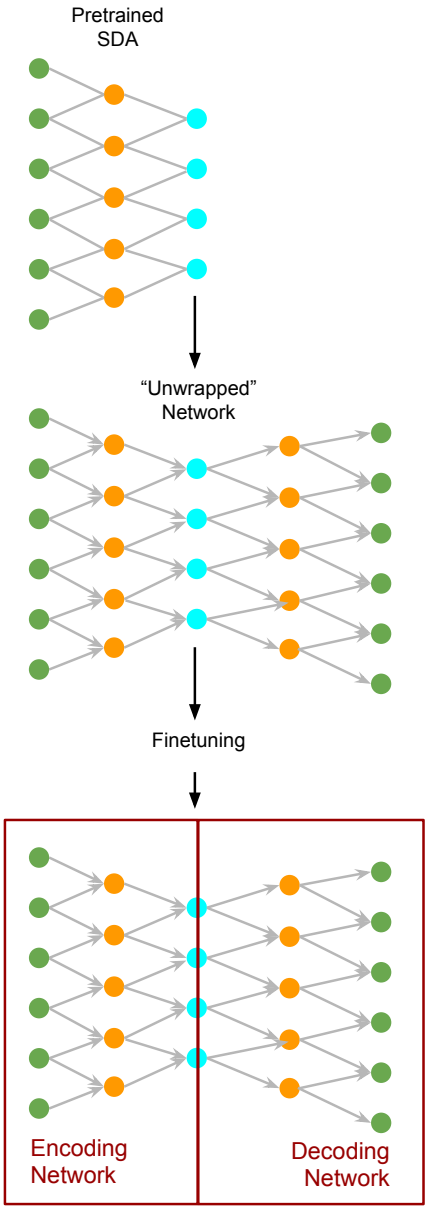


Figure 8: *Encoding and Decoding networks*

The unwrapped SDA acts as the initialization for a multilayer perceptron (MLP). An N layer SDA will produce an MLP with $2N - 1$ layers. Backpropagation is used to finetune the MLP such that the output layer can reconstruct the input provided to the first layer through the bottleneck in the middle. Once this finetuning has been completed, this network is split down the middle into two parts. The section from the input layer to the bottleneck region is the *encoding* network,

while the section from the bottleneck region to the output layer is the *decoding* network. The encoding network codes the speech signal into a representation which is by design, invertible, robust to noise, and low-dimensional. This representation is the encoding that the synthesizer uses as the parameterization of the speech signal i.e. it learns a mapping between the phonemes of text and the values of this encoding. At synthesis time, the synthesizer predicts values of this encoding based on the input text. The decoding network converts this code back into a representation of the speech signal. This approach is similar to the one proposed for efficient speech coding in [10]. Apart from the fact that [10] proposes the use of the code for other applications, it is also different in that it specifically looks for a binary encoding. Such binary encodings are not very useful in a statistical synthesis framework because binary representations are not interpolable while synthesis is an inherently generative task.

We have discussed how a deep neural network will build a low-dimensional noise-robust representation of the speech signal, but what should our deep neural network actually encode? To put it more explicitly, what should be the input to our deep neural network that it can learn to reconstruct? Should it be the actual speech signal itself, the magnitude spectrum, the complex spectrum, or any of the myriad other representations that signal processing research has provided us? In theory, the input representation should not matter since it has been proven that multilayer feedforward networks are universal approximators[26]. However, this proof places no constraints on the size or structure of the network. Nor does it provide a training algorithm that reaches the global optimum. Therefore, it is sensible to train the network on a representation that is known to be highly correlated with speech perception. Human hearing is known to be logarithmic both in amplitude[42], and frequency[47]. So, we propose that the Mel Spectrum and Mel Log spectrum are the most suitable representations that the network can be trained on. Despite using input and output layers that were linear, the network had difficulty working with the wide range of values in the Mel spectrum. Therefore, we will only describe our attempts at using a deep neural network to get an invertible, low-dimensional, noise-robust representation of the Mel Log Spectrum.

EXPERIMENTS AND RESULTS

The SDAs and the MLPs were built using the Theano[6] python library, and the parametric Speech Synthesizer using ClusterGen[7]. The input to the neural network was a 257-dimensional Mel Log Spectral vector which was obtained from a 512-point FFT of a 25ms speech frame. The encoding obtained using the network is 50-dimensional. This encoding size was chosen to make it easier for us to compare the

quality with the 50-dimensional MCEP representation used in our baseline system. The Stacked Denoising Autoencoder was built in a 257 x 125 x 75 x 50 configuration i.e. 257 nodes in the input layer, 125 in the first hidden layer, 75 in the second, and 50 in the output layer. This results in an MLP with a 257 x 125 x 75 x 50 x 75 x 125 x 257 configuration for fine-tuning. The encoding network will therefore have a configuration of 257 x 125 x 75 x 50 and the decoding network, 50 x 75 x 125 x 257. In all of these networks, the layer that is contact with the Mel Log Spectra is a linear layer with no non-linear function involved. This is so that the layer can deal with the range of values that the Mel Log Spectra can take. In all other layers, the neurons have sigmoid activations.

Experiments were run on 3 different voices: the RMS, and SLT voices from the CMU Arctic databases[32] and the Hindi corpus released as part of the 2014 Blizzard challenge[2]. The intention was to test the setup across gender as well as across language.

Evaluating the quality of the systems that we built poses an interesting problem. The standard objective metric used in nearly all evaluations of parametric speech synthesis is Mel Cepstral Distortion[33]. However, this metric is inherently unfair to our technique. This is because the default system that we compare against, like most statistical parametric synthesizers, works directly with the MCEPs. These systems optimize for the root-mean-squared error of MCEP prediction. In other words, they directly optimize for the metric. The technique that we are proposing gives us an encoding which is optimized for parametric synthesis. But optimizing for the prediction of this encoding need not necessarily optimize the Mel Cepstral Distortion directly. Therefore, any MCD-based results presented in this paper must be taken with a pinch of salt.

While our synthesizer might not directly optimize for MCD, the MCD is nevertheless a good indicator of listener perception; the argument here being that natural-sounding speech should have natural-appearing Mel Cepstral parameters. So, we will measure the quality of synthesis using the Mel Cepstra obtained from the Mel Log Spectra of the decoding network.

The first test is a simple analysis-resynthesis test. We measure how well our learned encoding is able to reconstruct the Mel Log Spectra of held-out test data. The results are shown in Table 2

Table 2: *Analysis-Resynthesis*

Voice	MCD Scores
ARCTIC RMS	4.354
ARCTIC SLT	4.315
Hindi corpus	3.916

In all of the three cases, the deep neural network trained for the voice is able to reconstruct the test set with relatively low error. It is however not obvious what these numbers should be compared against.

SYNTHESIS TESTS

The next set of tests were on using the deep neural network's 50-dimensional encoding as a parameterization for the ClusterGen statistical parametric synthesizer. MCD scores for the three above described voices are shown in Table 3.

Table 3: *ClusterGen synthesis voice build*

Voice	DNN Params	MCEP params
ARCTIC RMS	5.851	5.161
ARCTIC SLT	5.466	4.858
Hindi corpus	4.680	4.134

The Mel Cepstral Distortion is higher for the deep neural network encoding compared to the default system. In addition to this, the baseline system was preferred in informal subjective tests. As we had mentioned earlier in this section, we believe that the MCD of the DNN systems was affected by the fact that the Deep Neural Network was not directly optimizing for the score like the default system was doing. The lack of a good objective metric that would work with the DNN approach to parameterization exacerbated the problem by making it difficult to make design decisions. This inturn prevented us from making use of the full capability of the deep neural network; this is probably the reason for the lower subjective quality. We believe that the use of a better objective metric would reflect a more positive light on our results. It would also help us make better decisions which would contribute towards better parameterizations and improved subjective results.

These results are actually quite promising because the relatively good MCD scores we get with the DNN encoding strongly indicate that the encoding exists in an *interpolable* space. This is important because synthesizers like ClusterGen form clusters of the data vectors at the leaves of the trees and represent the cluster by its mean[57]. Therefore, only representations like MCEPs or Line Spectral Pairs[28] have been found to be suitable. The interpolable space constraint is probably the most difficult to achieve of the four earlier stated constraints. Even if our data-driven parameterization currently does slightly worse compared to MCEPs, it is extremely encouraging to be able to find that this parameterization manages to satisfy all of the

four requirements. Considering how close the difference is between the performance of the data-driven parameterization and that of the MCEPs, we expect that a more judicious design of the neural network coupled with better learning strategies will lead to great results in the future.

7 PROPOSED WORK

While we were successful in showing that a deep neural network could create a representation of the Mel Spectrum that satisfied all the constraints of synthesis, we have not yet used the neural network to its fullest capacity. The primary strength of the deep neural network is its ability to model highly non-linear functions. One fundamental problem in acoustic modeling in most modern synthesizers is that the vocal tract filter and the excitation source are modeled completely independent of each other. This assumption is wrong both from a physiological point of view as well as a computational one. This is fairly well known in the speech community but attempts at creating joint representations of the vocal tract and source models in the past have been fairly unsuccessful.

I therefore propose to apply the technique described in Chapter 6 to try to create a joint representation of the source signal and the vocal tract filter. The setup will be as shown in Figure 9. The input at the first layer will be a representation of the vocal tract such as STRAIGHT parameters or the Mel Spectrum *and* the parameters of the source model. The deep neural network will then learn a joint representation which is designed to be invertible, low-dimensional, noise-robust, and possibly interpolable.

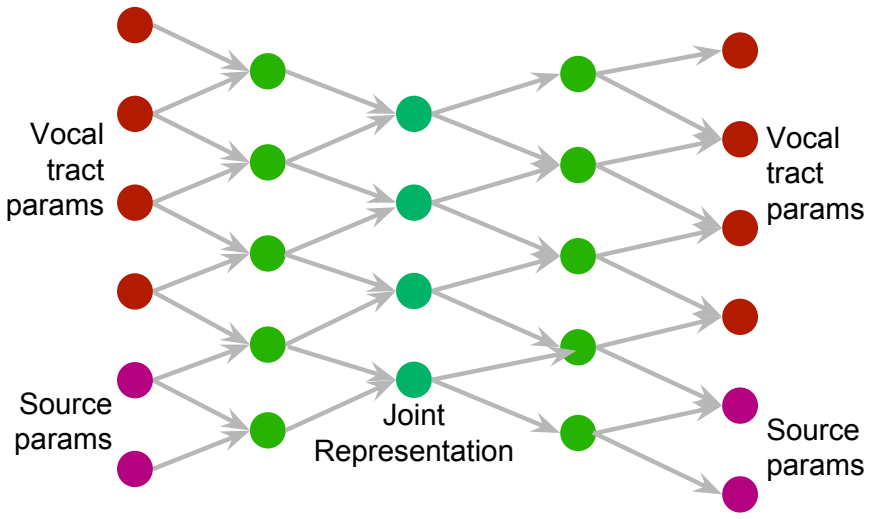


Figure 9: *Joint Modeling of source and vocal tract*

If this technique succeeds, it will be particularly useful in dealing with source models such as [12]. This source model provides near perfect reconstruction when coupled with LSPs as a vocal tract representation. Unfortunately, the model parameters are of relatively high dimension and were not very noise robust in the ClusterGen framework. We believe that the deep neural network might help us create lower dimensional, noise-robust versions of these parameters.

INCORPORATING LONGER TERM FEATURES

Nearly all modern synthesizers are *concatenative* synthesizers meaning that the speech is synthesized in small chunks and then concatenated together to form coherent speech. In Unit Selection synthesizers, speech is created by concatenating phones, diphones, or other slightly longer units. In Parametric synthesis, speech is created by concatenating model parameters and then synthesizing them using the parametric model. Humans, on the other hand, do not produce speech one frame, one phone, or one diphone at a time. The human speech production mechanisms are always aware of the full context of the sound being produced, while synthesizers have at most knowledge of the sounds immediately preceding and following the current sound. Even when this knowledge is available, the knowledge is purely phonetic. The synthesizer is practically unaware of the dynamics of the signal over longer periods of time. Speech Recognition systems attempt to partially rectify this problem by *stacking* features i.e. creating 'superframes' which contain 5 or 6 frames of speech stacked together. Principal Component Analysis[20] or Linear Discriminant Analysis[38] is then used to reduce the number of dimensions. Unfortunately, these are not invertible transformations and so cannot be used for speech synthesis.

Other techniques such as Frequency Domain Linear Prediction[4] and Wavelet[48] based methods work well in producing a relatively low dimensional representation of the longer term dynamics of the signal but incorporating them into modern synthesizers has proven to be difficult.

I hypothesize that the deep neural network based technique described earlier could also be used to incorporate longer term dynamics into the parameterization that is created. The exact structure of the network that could be used for this purpose requires a bit of thought and a great deal of experimentation. So, at this stage, I will avoid making any explicit claims.

Part IV

IMPROVED EVALUATIONS

8

BACKGROUND AND PREVIOUS WORK

The astute reader will have noticed that the objective metrics used to evaluate the performance of the techniques described in Part ii and Part iii were not ideal for the task at hand. For the excitation model, we used Root-Mean-Squared-Error and Correlation for the fitting process. These metrics are heavily influenced by changes in the magnitude of the synthesized and reference signals. An added problem is that these metrics are also non-standard. So, the results published in our papers cannot be compared to earlier systems easily. The correlation between these metrics and perceptual quality is also unclear. So, it is difficult to quantify the amount of improvement in the metric necessary for a noticeable improvement in synthesis quality. Commonly used metrics like Mel Cepstral Distortion[33] or Line Spectral Distortion measure aspects of the speech signal which are more or less orthogonal to the excitation model. So, improvements or degradations in the excitation model will not be reflected in these.

In the deep learning models, we had the problem that our evaluation metric, MCD was heavily biased towards the baseline system since this is the metric the baseline system optimizes for. Perceptual quality, on the other hand, is actually better correlated with measurements in the Mel Spectral domain rather than the Mel Cepstral domain. Yet, since the objective metric is MCD, it hurts the system to optimize for Mel Spectral distortion.

But the major shortcoming of Mel Cepstral Distortion and nearly all other objective metrics used for evaluating speech synthesis today is that they are fundamentally speech *coding* metrics rather than speech *synthesis* metrics. In fact, a quick perusal through a list of modern objective speech quality metrics[39] will reveal that there really is *no* measure designed with synthesis in mind. This is particularly obvious when looking at metrics like PESQ[41] which is a fairly sophisticated measure but most of the complexity arises due to processes like time aligning the reference and test signal. This is unnecessary for speech synthesis since we know exactly where each phone exists in both the reference and test signals. Excluding all of that, the underlying metric of distortion in PESQ is just Bark Spectral Distortion which is not significantly different from the Mel Spectral Distortion measure currently used in synthesis.

But why is this the case? Well, every metric listed in [39] assumes that there exists a reference speech signal which the decoded or synthesized speech signal is compared to. While this makes perfect sense for coding tasks like telephony, we really must ask ourselves if a ref-

erence is necessary for evaluating the quality of speech synthesis. If a speech synthesizer were to say the sound 'pa', all we really care about is that the synthesized speech sound like a 'pa' uttered by the appropriate speaker. It does not matter whether the two waveforms match exactly down to the bit level as long as it sounds right to a human. In fact, it is actually impossible to get the synthesized speech to match the reference exactly for sounds like fricatives. These sounds have characteristics very close to white noise and therefore the synthesizer will never be able to reproduce this perfectly. While an objective metric which does not need a reference signal is highly desirable, it is also extremely difficult to create such a metric. But it is reasonable to at least expect that the metric take into account the sound being produced when computing the distance between the reference and the synthesized speech signal.

One other fundamental flaw in most objective metrics today is the tendency to measure distance at the frame level with little or no context. For instance, let us consider the case of MCD and for clarity, we shall use the simple example of measuring distortion on just one coefficient for each frame. A plot of the true values of this hypothetical parameter across time are shown in Figure 10.

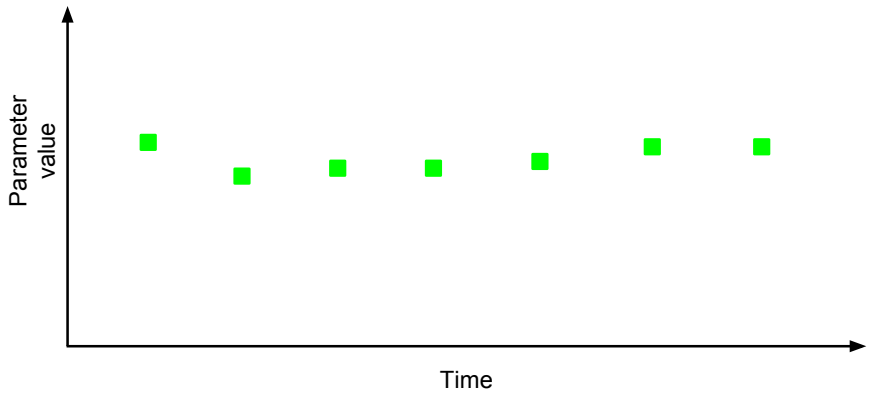


Figure 10: *True values of the parameter*

Let us now consider two possible sets of predictions of this parameter. Our task is to decide between the set of predictions in Figure 11 shown in red and the set of predictions in Figure 12 shown in purple.

These two sets of parameters have identical MCD scores. However, any speech researcher would immediately point out that the parameter values in purple are better. The reason for this being that large differences between consecutive parameter values would cause a lot of distortions in the synthesized speech. Since the MCD calculation does not take into account distortions across frames, errors like these will go unnoticed. While it may appear that switching to a distortion metric that is based on the delta parameters (differences between pa-

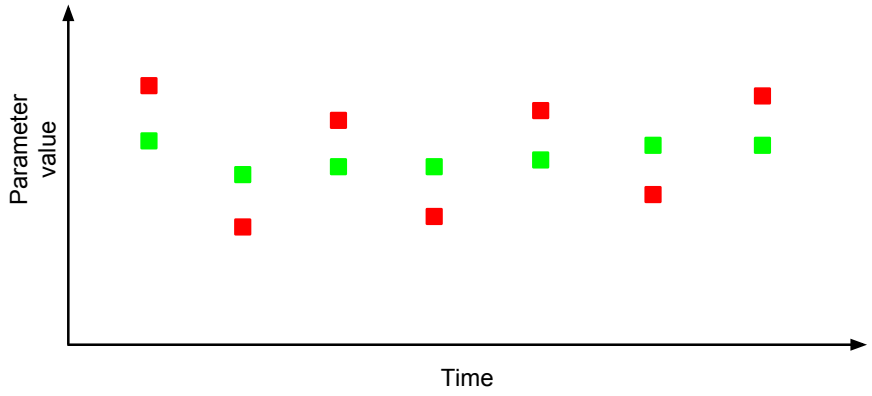


Figure 11: *Predicted parameters set 1*

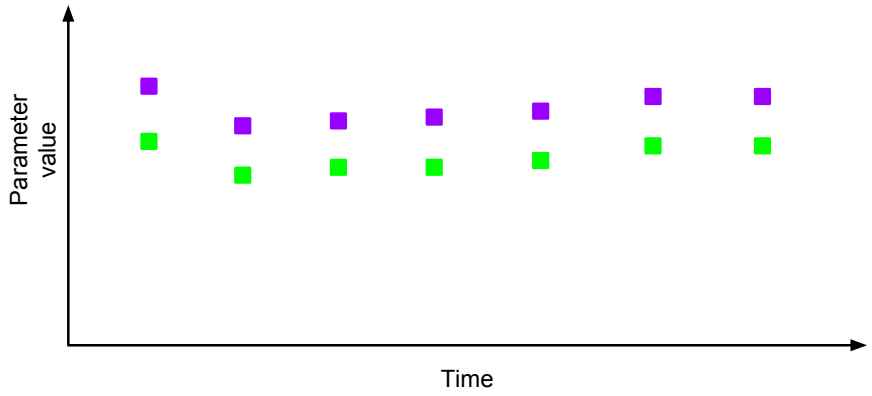


Figure 12: *Predicted parameters set 2*

rameters of consecutive frames) might fix this, relying on deltas is still a poor fix since the trajectory of the signal over intervals longer than two frames will still be ignored. Instead, what we need is a composite metric that can take into account both short and long term dynamics of the speech signal.

9 PROPOSED WORK

In the last few years, there has been a significant amount of interest in the synthesis community in the use of post-filtering techniques that modify the longer term dynamics of the signal. One such method is the Modulation Spectrum technique proposed in [50]. This technique looks at the trajectories of individual Mel Cepstral Coefficients across several frames of an utterance. The Fourier transform of each of these trajectories is then computed resulting in a *Modulation Spectrum* or MS. The MS for a each phone is then modeled using a Gaussian Mixture Model. On the synthesis end, this GMM is used to enforce the parameter to have a particular Modulation Spectrum i.e. a particular trajectory across time. In the paper, subjective tests indicate that the enforcing this trajectory results in more natural sounding synthesis. What is more interesting for this particular discussion though is that despite obvious improvements in subjective quality, enforcing this trajectory results in a *higher* (worse) MCD score.

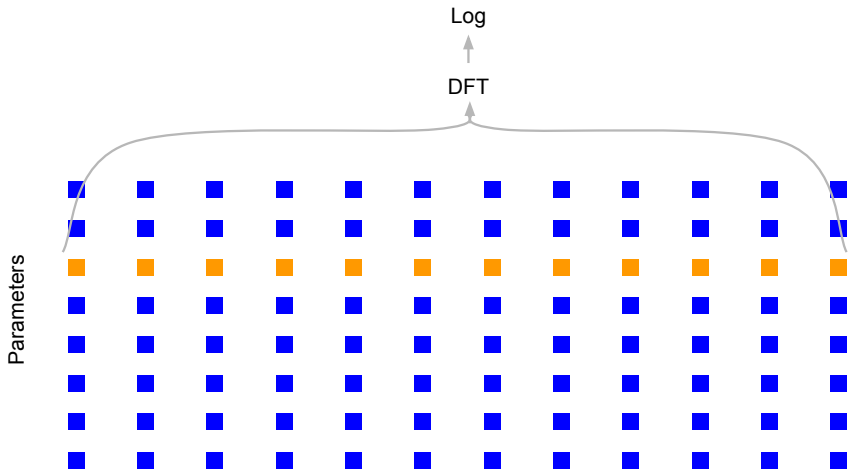


Figure 13: *Modulation Spectrum computation*

It is apparent that the long term dynamics of the signal matter quite a lot and that MCD does not adequately measure these. We must therefore consider using metrics that take longer term dynamics into account. One option is to use the Modulation Spectrum itself. Measuring distortion in the MS domain might be able to detect the distortions that occur in the long term trajectories of the individual parameters across multiple frames. The MS technique is quite similar to the Frequency-Domain Linear Prediction described in [4]; the main difference between the two is that the latter codes the parameters

using LPC rather than the log DFT. It is difficult to predict which of these two techniques will be better suited for evaluating speech synthesis. However, considering the ease of implementation of these two techniques, I plan to investigate the use of both. It may very well turn out that the best evaluation metric is a combination of *both* of these techniques in addition to the traditional frame-based distortion measurements.

SPECTROTEMPORAL RECEPTIVE FIELDS

While the use of Frequency-Domain Linear Prediction and the Modulation Spectrum might be a major shift from the conventional ways of measuring distortion in speech synthesis, these are still metrics which could be considered speech coding metrics and there is nothing in them which is inherently specific to speech synthesis.

To get an idea of why a speech synthesis specific metric is necessary, let's take a look at the spectrogram shown in Figure 14, specifically at the section inside the red box.

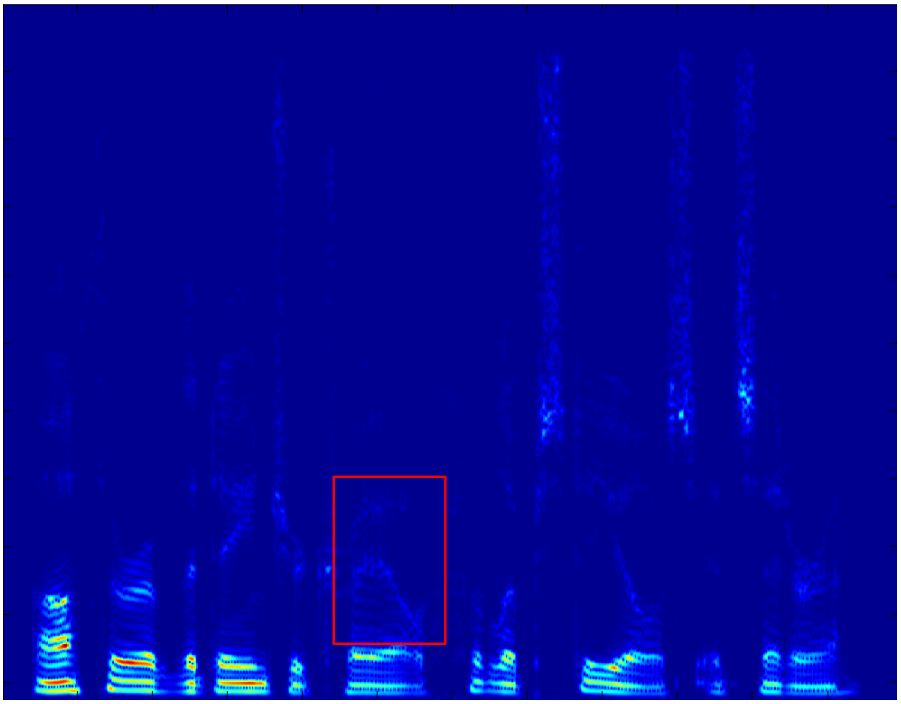


Figure 14: *"Author of the danger trail, Philip Steels, etc"*

The box highlights the formants for the diphthong in the word 'trail'. Diphthongs are fundamentally characterized by formant trajectories that extend across multiple frequencies in a relatively short period of time. Neither MCD nor any of the longer term metrics that I have proposed in the previous section are capable of accurately detecting trajectories that extend both across time and frequency. We must therefore move away from naive metrics that only consider the

parameters in one dimension and treat speech as the two dimensional signal that it actually is. One reason why this might work for synthesis but not for coding is that we know the exact *boundaries* of every phone in the utterances we synthesize. We could therefore apply two dimensional representations like these only to the phones where it matters. It makes a lot of sense to analyze two-dimensional representations in diphthongs, but makes no sense to do so in fricatives.

There are also physiologically motivated reasons for using two dimensional representations of the speech signal. Studies of the primary auditory cortex of small mammals have found that the responses of the neurons correspond to joint spectro-temporal patterns. In other words, the neurons in the auditory cortex fire in response to specific two dimensional patterns in the spectrogram such as upward or downward moving ripples rather than just one-dimensional patterns. The specific patterns that the neurons respond to are called Spectro-Temporal Receptive Fields (STRFs for short). A Receptive Field is a particular pattern or stimulus that causes a neuron to fire, and each STRF is a specific spectro-temporal pattern that will cause the neuron to fire. Detailed studies of these experiments are presented in [11] and [44].

Many of these STRFs can be closely approximated using two dimensional Gabor filters. A Gabor Filter is the product of a Gaussian envelope and a sinusoid. Figure 15 and Figure 16 show two different views of a 2D Gabor filter.

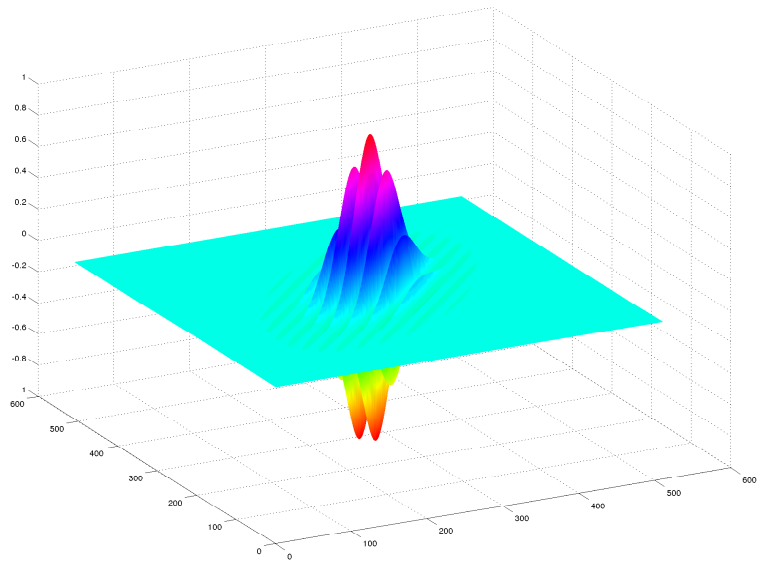


Figure 15: *2D Gabor filter*

Convolution of this filter with the spectrogram of the speech signal will result in detection of all downward moving ripples. By using

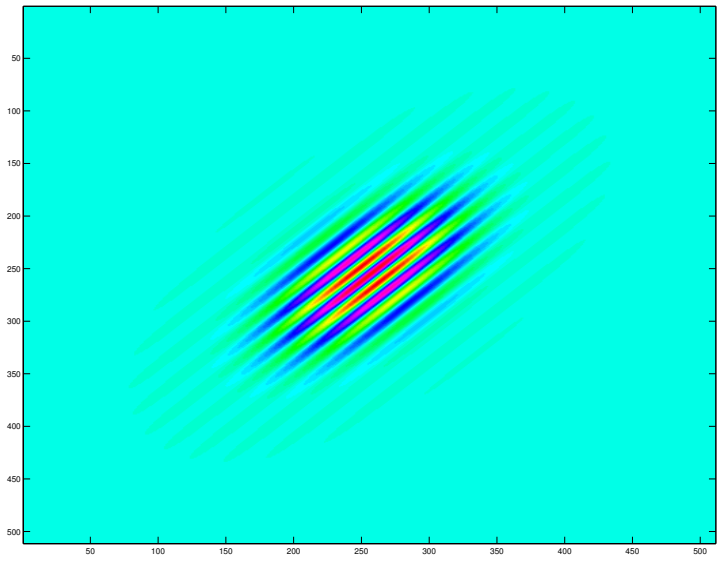


Figure 16: *2D Gabor filter top view*

a multitude of such Gabor filters with varying sinusoid frequencies, orientations, and scales, the formant trajectories in the speech signal can be detected in the speech signal. These features can then be used as an evaluation metric for speech synthesis.

At this stage, it is unclear what the best way to measure distance using these features would be. It could be a simple L2 distance or more complex regression techniques. It might also be necessary to apply dimensionality-reduction techniques to the features. At any rate, the answers to all of these questions should become more obvious once the research progresses.

As I had mentioned earlier, the use of STRFs for evaluating speech synthesis is strongly motivated by my belief that evaluation metrics should listen to the speech the same way humans do. There is a large body of literature dedicated to understanding the human auditory system (an overview of these techniques is available here:[46]). While it will be impossible for me to implement all of these models, I plan to try using all auditory models for which *working* code is available. Some examples of these are Power Normalized Cepstral Coefficients[31], the Slaney auditory toolbox[45] and the code from the Carney lab such as the one for this paper:[58].

All of the proposed experiments in this section will be run on the Blizzard challenge[1] evaluation data which contains synthesis examples from many different systems over the years and the corresponding ratings that human listeners assigned to these.

Part V

CONCLUSIONS AND TIMELINE

10 SUMMARY OF PROPOSED WORK

In the previous chapters, we have discussed the various problems that plague the acoustic models of speech synthesis; my work towards fixing some of these problems; and the proposed work that I plan to do for the rest of my thesis. In this chapter, I will paraphrase the proposed work presented in all the earlier chapters. This consolidated chapter is purely for the convenience of my thesis committee, and does not contain any information that doesn't already exist in earlier chapters.

In Part ii, I described the problems with the source models used in current synthesizers. I will try to mitigate the problems caused by imperfect source models by using a hybrid approach using both parametric and non-parametric models to represent the glottal source signal. The parametric model used will probably be the Liljencrants-Fant model. The non-parametric model used will be decided later based on literature review and experiments.

In Part iii, I had explained the necessity of creating representations specific to the goals of synthesis. I believe that such representations can be created using Deep learning approaches. I will investigate the use of these techniques in creating joint models of the vocal tract and the source characteristics.

In Part iv, I had highlighted the shortcomings of current objective metrics especially when measuring aspects of the speech signal that will be improved by the improved techniques of the previous chapters. I propose the use of features that span longer intervals of time and frequency, auditory models, and joint time-frequency features as a potential source for creating better objective metrics.

11 TIMELINE

July-Aug, 2014	Work on improving objective Metrics
Sep 2014	Re-examine neural network training process to consider features that provided better objective metrics
Oct-Nov, 2014	Work on improving excitation models
Dec 2014	Re-assess neural network training to add excitation models to the process
Jan-Feb 2015	Re-implement neural networks after factoring in decisions made earlier
Mar 2015	Re-design objective metrics based on earlier results
Apr 2015	Re-design excitation models based on earlier results
May-June 2015	Consolidate and implement final versions
July-Aug 2015	Thesis writing
Aug 2015	Thesis Defense

BIBLIOGRAPHY

[1] Blizzard challenge. http://festvox.org/blizzard/index.html, . (Cited on pages 8, 19, and 40.)

[2] Blizzard challenge 2014. http://www.synsig.org/index.php/Blizzard_Challenge_2014, . (Cited on page 25.)

[3] Paavo Alku. Glottal wave analysis with pitch synchronous iterative adaptive inverse filtering. *Speech communication*, 11(2): 109–118, 1992. (Cited on page 9.)

[4] Marios Athineos and Daniel PW Ellis. Frequency-domain linear prediction for temporal features. In *Automatic Speech Recognition and Understanding, 2003. ASRU'03. 2003 IEEE Workshop on*, pages 261–266. IEEE, 2003. (Cited on pages 30 and 37.)

[5] Yoshua Bengio, Pascal Lamblin, Dan Popovici, Hugo Larochelle, et al. Greedy layer-wise training of deep networks. *Advances in neural information processing systems*, 19:153, 2007. (Cited on page 21.)

[6] James Bergstra, Olivier Breuleux, Frédéric Bastien, Pascal Lamblin, Razvan Pascanu, Guillaume Desjardins, Joseph Turian, David Warde-Farley, and Yoshua Bengio. Theano: a CPU and GPU math expression compiler. In *Proceedings of the Python for Scientific Computing Conference (SciPy)*, June 2010. (Cited on page 24.)

[7] Alan W Black. ClusterGen: a statistical parametric synthesizer using trajectory modeling. In *INTERSPEECH*, 2006. (Cited on pages 12 and 24.)

[8] Leo Breiman, Jerome H Friedman, Richard A Olshen, and Charles J Stone. Classification and regression trees. 1998. (Cited on page 12.)

[9] Gang Chen, Marc Garellek, Jody Kreiman, Bruce R Gerratt, and Abeer Alwan. A perceptually and physiologically motivated voice source model. In *Interspeech*, pages 2001–2005. ISCA, 2013. (Cited on page 15.)

[10] Li Deng, Michael L Seltzer, Dong Yu, Alex Acero, Abdel-Rahman Mohamed, and Geoffrey E Hinton. Binary coding of speech spectrograms using a deep auto-encoder. In *Interspeech*, pages 1692–1695. ISCA, 2010. (Cited on page 24.)

[11] Didier A Depireux, Jonathan Z Simon, David J Klein, Shihab A Shamma, et al. Spectro-temporal response field characterization with dynamic ripples in ferret primary auditory cortex. *Journal of neurophysiology*, 85(3):1220–1234, 2001. (Cited on page 39.)

[12] Thomas Drugman, Geoffrey Wilfart, and Thierry Dutoit. A deterministic plus stochastic model of the residual signal for improved parametric speech synthesis. In *INTERSPEECH*, pages 1779–1782, 2009. (Cited on pages 8, 16, and 30.)

[13] Thierry Dutoit and Bernard Gosselin. On the use of a hybrid harmonic/stochastic model for TTS synthesis-by-concatenation. *Speech Communication*, 19(2):119–143, 1996. (Cited on page 19.)

[14] Thierry Dutoit and Henri Leich. MBR-PSOLA text-to-speech synthesis based on an MBE re-synthesis of the segments database. *Speech Communication*, 13(3):435–440, 1993. (Cited on page 19.)

[15] Gunnar Fant. *Acoustic theory of speech production*. Walter de Gruyter, 1970. (Cited on pages 3 and 7.)

[16] Gunnar Fant. Glottal source and excitation analysis. *Speech Transmiss. Lab. Q. Prog. Stat. Rep*, 1:70–85, 1979. (Cited on page 8.)

[17] Gunnar Fant. Some problems in voice source analysis. *Speech Communication*, 13(1):7–22, 1993. (Cited on page 8.)

[18] Gunnar Fant. The voice source in connected speech. *Speech communication*, 22(2):125–139, 1997. (Cited on page 8.)

[19] Gunnar Fant, Johan Liljencrants, and Qi-guang Lin. A four-parameter model of glottal flow. *STL-QPSR*, 4(1985):1–13, 1985. (Cited on page 9.)

[20] Ronald A Fisher. The use of multiple measurements in taxonomic problems. *Annals of eugenics*, 7(2):179–188, 1936. (Cited on page 30.)

[21] Hiroya Fujisaki and Mats Ljungqvist. Proposal and evaluation of models for the glottal source waveform. In *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP'86.*, volume 11, pages 1605–1608. IEEE, 1986. (Cited on pages 8 and 15.)

[22] Jonas Gehring, Yajie Miao, Florian Metze, and Alex Waibel. Extracting deep bottleneck features using stacked auto-encoders. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 3377–3381. IEEE, 2013. (Cited on page 22.)

[23] Geoffrey Hinton, Li Deng, Dong Yu, George E Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Tara N Sainath, et al. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *Signal Processing Magazine, IEEE*, 29(6):82–97, 2012. (Cited on page 22.)

[24] Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313(5786): 504–507, 2006. (Cited on page 21.)

[25] Geoffrey E Hinton, Simon Osindero, and Yee-Whye Teh. A fast learning algorithm for deep belief nets. *Neural computation*, 18 (7):1527–1554, 2006. (Cited on page 21.)

[26] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366, 1989. (Cited on page 24.)

[27] Andrew J Hunt and Alan W Black. Unit selection in a concatenative speech synthesis system using a large speech database. In *Acoustics, Speech, and Signal Processing, 1996. ICASSP-96. Conference Proceedings., 1996 IEEE International Conference on*, volume 1, pages 373–376. IEEE, 1996. (Cited on page 3.)

[28] F Itakura. Line spectrum representation of linear predictor coefficients of speech signals. *Acoustical Society of America Journal*, 57: 35, 1975. (Cited on pages 13 and 26.)

[29] Hideki Kawahara. STRAIGHT, exploitation of the other aspect of VOCODER: Perceptually isomorphic decomposition of speech sounds. *Acoustical science and technology*, 27(6):349–353, 2006. (Cited on page 8.)

[30] Hideki Kawahara, Masanori Morise, Toru Takahashi, Ryuichi Nisimura, Toshio Irino, and Hideki Banno. TANDEM-STRAIGHT: A temporally stable power spectral representation for periodic signals and applications to interference-free spectrum, fo, and aperiodicity estimation. In *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*, pages 3933–3936. IEEE, 2008. (Cited on pages 8 and 19.)

[31] Chanwoo Kim and Richard M Stern. Power-normalized cepstral coefficients (pncc) for robust speech recognition. In *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*, pages 4101–4104. IEEE, 2012. (Cited on page 40.)

[32] John Kominek and Alan W Black. The cmu arctic speech databases. In *Fifth ISCA Workshop on Speech Synthesis*, 2004. (Cited on pages 14 and 25.)

[33] Robert F Kubichek. Mel-cepstral distance measure for objective speech quality assessment. In *Communications, Computers and Signal Processing, 1993., IEEE Pacific Rim Conference on*, volume 1, pages 125–128. IEEE, 1993. (Cited on pages 25 and 33.)

[34] Hugo Larochelle, Yoshua Bengio, Jérôme Louradour, and Pascal Lamblin. Exploring strategies for training deep neural networks. *The Journal of Machine Learning Research*, 10:1–40, 2009. (Cited on page 21.)

[35] Eric Moulines and Francis Charpentier. Pitch-synchronous waveform processing techniques for text-to-speech synthesis using diphones. *Speech communication*, 9(5):453–467, 1990. (Cited on page 19.)

[36] Prasanna Kumar Muthukumar, Alan W Black, and H. Timothy Bunnell. Optimizations and fitting procedures for the liljencrants-fant model in statistical parametric speech synthesis. In *Proceedings of INTERSPEECH*, 2013. (Cited on page 9.)

[37] Alok Parlikar. TestVox: Web-based Framework for Subjective Evaluation of Speech Synthesis. Opensource Software, 2012. (Cited on page 13.)

[38] Karl Pearson. Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572, 1901. (Cited on page 30.)

[39] Schuyler R Quackenbush, Thomas Pinkney Barnwell, and Mark A Clements. *Objective measures of speech quality*. Prentice Hall Englewood Cliffs, NJ, 1988. (Cited on page 33.)

[40] Tuomo Raitio, Antti Suni, Junichi Yamagishi, Hannu Pulakka, Jani Nurminen, Martti Vainio, and Paavo Alku. Hmm-based speech synthesis utilizing glottal inverse filtering. *Audio, Speech, and Language Processing, IEEE Transactions on*, 19(1):153–165, 2011. (Cited on page 9.)

[41] Antony W Rix, John G Beerends, Michael P Hollier, and Andries P Hekstra. Perceptual evaluation of speech quality (pesq)-a new method for speech quality assessment of telephone networks and codecs. In *Acoustics, Speech, and Signal Processing, 2001. Proceedings.(ICASSP'01). 2001 IEEE International Conference on*, volume 2, pages 749–752. IEEE, 2001. (Cited on page 33.)

[42] Derek W Robinson and R So Dadson. A re-determination of the equal-loudness relations for pure tones. *British Journal of Applied Physics*, 7(5):166, 1956. (Cited on page 24.)

[43] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. *Learning representations by back-propagating errors*. MIT Press, Cambridge, MA, USA, 1988. (Cited on page 21.)

[44] Christoph E Schreiner and Barbara M Calhoun. Spectral envelope coding in cat primary auditory cortex: properties of ripple transfer functions. *Aud Neurosci*, 1(1):39–62, 1994. (Cited on page 39.)

[45] Malcolm Slaney. Auditory toolbox. *Interval Research Corporation, Tech. Rep*, 10:1998, 1998. (Cited on page 40.)

[46] Richard M Stern and Nelson Morgan. Features based on auditory physiology and perception. *Techniques for Noise Robustness in Automatic Speech Recognition*, pages 193–227, 2012. (Cited on page 40.)

[47] SS Stevens, J Volkmann, and EB Newman. A scale for the measurement of the psychological magnitude pitch. *Journal of the Acoustical Society of America*, 1937. (Cited on page 24.)

[48] Gilbert Strang and Truong Nguyen. *Wavelets and filter banks*. SIAM, 1996. (Cited on page 30.)

[49] Yannis Stylianou. Applying the harmonic plus noise model in concatenative speech synthesis. *Speech and Audio Processing, IEEE Transactions on*, 9(1):21–29, 2001. (Cited on pages 8 and 19.)

[50] Shinnosuke Takamichi, Tomoki Toda, Graham Neubig, Sakriani Sakti, and Satoshi Nakamura. A postfilter to modify the modulation spectrum in HMM-based speech synthesis. In *Acoustics, Speech and Signal Processing, 2014. ICASSP 2014. IEEE International Conference on*. IEEE, 2014. (Cited on page 37.)

[51] Tomoki Toda and Keiichi Tokuda. A speech parameter generation algorithm considering global variance for HMM-based speech synthesis. *IEICE TRANSACTIONS on Information and Systems*, 90(5):816–824, 2007. (Cited on page 19.)

[52] Keiichi Tokuda, Takao Kobayashi, Takashi Masuko, and Satoshi Imai. Mel-generalized cepstral analysis-a unified approach to speech spectral estimation. In *ICSLP*, 1994. (Cited on page 19.)

[53] Keiichi Tokuda, Takao Kobayashi, and Satoshi Imai. Speech parameter generation from HMM using dynamic features. In *Acoustics, Speech, and Signal Processing, 1995. ICASSP-95., 1995 International Conference on*, volume 1, pages 660–663. IEEE, 1995. (Cited on page 19.)

[54] Damien Vincent, Olivier Rosec, and Thierry Chonavel. A new method for speech synthesis and transformation based on an

ARX-LF source-filter decomposition and HNM modeling. In *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, volume 4, pages IV–525. IEEE, 2007. (Cited on page 12.)

[55] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning*, pages 1096–1103. ACM, 2008. (Cited on page 21.)

[56] Takayoshi Yoshimura, Keiichi Tokuda, Takashi Masuko, Takao Kobayashi, and Tadashi Kitamura. Mixed excitation for hmm-based speech synthesis. In *INTERSPEECH*, pages 2263–2266, 2001. (Cited on pages 8 and 13.)

[57] Heiga Zen, Keiichi Tokuda, and Alan W Black. Statistical parametric speech synthesis. *Speech Communication*, 51(11):1039–1064, 2009. (Cited on pages 3, 19, and 26.)

[58] Xuedong Zhang, Michael G Heinz, Ian C Bruce, and Laurel H Carney. A phenomenological model for the responses of auditory-nerve fibers: I. nonlinear tuning with compression and suppression. *The Journal of the Acoustical Society of America*, 109 (2):648–670, 2001. (Cited on page 40.)

COLOPHON

This document was typeset using the typographical look-and-feel `classicthesis` developed by André Miede. The style was inspired by Robert Bringhurst's seminal book on typography "*The Elements of Typographic Style*". `classicthesis` is available for both LaTeX and LyX:

http://code.google.com/p/classicthesis/

Happy users of `classicthesis` usually send a real postcard to the author, a collection of postcards received so far is featured here:

http://postcards.miede.de/