FloWaveNet : A Generative Flow for Raw Audio

Sungwon Kim [1] Sang-gil Lee [1] Jongyoon Song [1] Jaehyeon Kim [2] Sungroh Yoon [1 3]

Abstract

Most modern text-to-speech architectures use a WaveNet vocoder for synthesizing high-fidelity waveform audio, but there have been limitations, such as high inference time, in its practical application due to its ancestral sampling scheme. The recently suggested Parallel WaveNet and ClariNet have achieved real-time audio synthesis capability by incorporating inverse autoregressive flow for parallel sampling. However, these approaches require a two-stage training pipeline with a well-trained teacher network and can only produce natural sound by using probability distillation along with auxiliary loss terms. We propose FloWaveNet, a flow-based generative model for raw audio synthesis. FloWaveNet requires only a single-stage training procedure and a single maximum likelihood loss, without any additional auxiliary terms, and it is inherently parallel due to the characteristics of generative flow. The model can efficiently sample raw audio in real-time, with clarity comparable to previous two-stage parallel models. The code and samples for all models, including our FloWaveNet, are publicly available.

1. Introduction

The end-to-end waveform audio synthesis model is a core component of text-to-speech systems. The striking success of deep learning based raw audio synthesis architectures has led to modern text-to-speech systems leveraging them as vocoders to synthesize realistic waveform signals that are nearly indistinguishable from natural sound in the real world.

Current state-of-the-art text-to-speech architectures commonly use the WaveNet vocoder with a mel-scale spectrogram as an input for high-fidelity audio synthesis (Shen et al., 2018; Arik et al., 2017b;a; Ping et al., 2017; Jia et al., 2018). However, the practical application of WaveNet has been limited because it requires an autoregressive sampling scheme, which serves as a major bottleneck in real-time waveform generation.

Several variations of the original WaveNet have been proposed to overcome its slow ancestral sampling. Parallel WaveNet (Van Den Oord et al., 2017) has achieved real-time audio synthesis by incorporating inverse autoregressive flow (IAF) (Kingma et al., 2016) for parallel audio synthesis. The recently suggested ClariNet (Ping et al., 2018) presented an alternative formulation by using a single Gaussian distribution with a closed-form Kullback-Leibler (KL) divergence, contrasting with the high-variance Monte Carlo approximation from Parallel WaveNet.

Despite the success of real-time high-fidelity waveform audio synthesis, all of the aforementioned approaches require a two-stage training pipeline with a well-performing pre-trained teacher network for a probability density distillation training strategy. Furthermore, in practical terms, these models can synthesize realistic audio samples only by using additional auxiliary losses. For example, if only probability density distillation loss is used, Parallel WaveNet is prone to mode collapse, in which the student network converges to a certain mode of the teacher distribution, resulting in sub-optimal performance (Van Den Oord et al., 2017).

Here, we present FloWaveNet, a flow-based approach that is an alternative to the real-time parallel generative model for raw audio synthesis. FloWaveNet requires only a single maximum likelihood loss, without any auxiliary loss terms, while maintaining stability in training. It features a simplified single-stage training scheme because it does not require a teacher network and can be trained end-to-end. The model is inherently parallel because of flow-based generation, which enables real-time waveform synthesis. FloWaveNet can act as a drop-in replacement for the WaveNet vocoder, which is used in a variety of text-to-speech architectures. Along with all the advantages described above, the quality and fidelity of samples from FloWaveNet are comparable to the two-stage models.

Currently, there is no official implementation of the aforementioned two-stage models available, and the performance of publicly accessible implementations does not match the

[1] Electrical and Computer Engineering, Seoul National University, Seoul, Korea [2] Kakao Corporation [3] ASRI, INMC, Institute of Engineering Research, Seoul National University, Seoul, Korea. Correspondence to: Sungroh Yoon <sryoon@snu.ac.kr>.

Proceedings of the 36th International Conference on Machine Learning, Long Beach, California, PMLR 97, 2019. Copyright 2019 by the author(s).

result reported in their respective papers. In addition to our FloWaveNet, we present an open source implementation of the Gaussian IAF model that outperforms current public implementations, along with the first comparative study with the previous parallel synthesis model on a publicly available speech dataset. The code and samples for all models, including FloWaveNet, are publicly available[1][2][3].

Our major contributions are as follows:

- We propose FloWaveNet, a new flow-based approach for parallel waveform speech synthesis which requires only a single maximum likelihood loss and an end-to-end single-stage training, in contrast to previous two-stage approaches.
- We show the difficulty of generating realistic audio with the two-stage approach without using the auxiliary loss terms, whereas the training of FloWaveNet is greatly simplified and stable throughout iterations.
- We present an open source implementation of FloWavenet and the Gaussian IAF that outperforms publicly available implementations, along with the first comparative study between methods using a publicly available speech dataset.

The rest of this paper is organized as follows: In Section 2.1, we provide a summary of the original WaveNet and the speed bottleneck for real-world applications. Section 2.2 provides core backgrounds (IAF and *probability density distillation*) of the recently proposed parallel speech synthesis method and we describe two previous works and their limitations. Section 3 presents our FloWaveNet model and Section 4 shows experimental details. We provide crowd-sourced mean opinion score (MOS) results and further analysis on the behaviors of each models in Section 5.

2. Related Work

2.1. WaveNet

WaveNet (Van Den Oord et al., 2016) is a generative model that estimates the probability distribution of raw audio, and it can synthesize state-of-the-art fidelity speech and audio. The model decomposes the joint probability distribution of the audio signal $x_{1:T}$ into a product of conditional probabilities as follows:

$$P_X(x_{1:T}) = \prod_t P_X(x_t|x_{<t}). \tag{1}$$

The model estimates the joint probability by using causal dilated convolutions, which can successfully model the long-term dependency of the conditional probabilities. The original WaveNet architecture used an 8-bit quantized signal and estimated the conditional probabilities via a 256-way categorical distribution by using a softmax function. Subsequent studies (Van Den Oord et al., 2017; Ping et al., 2018) replaced the categorical distribution with a discretized mixture of logistics (MoL) or single Gaussian distribution directly by using 16-bit audio for a higher-fidelity sound.

The WaveNet structure is a fully convolutional network without recurrence, which can enable an efficient parallel probability estimation for all time steps and parallel training, via a teacher-forcing technique. However, because of the autoregressive nature of the model, WaveNet can only use the ancestral sampling scheme, which runs a full forward pass of the model for each time step during inference, which is a major speed bottleneck in the audio synthesis. The official implementation generates 172 samples per second, which is approximately 140 times slower than real-time for 24kHz audio (Van Den Oord et al., 2017).

2.2. Parallel Speech Synthesis Models

The main objective of parallel speech synthesis models (Van Den Oord et al., 2017; Ping et al., 2018) is to endow the system with the ability to synthesize audio in parallel at all time steps during inference, while also ensuring sound fidelity and quality comparable to the original autoregressive WaveNet. To this end, parallel speech synthesis models incorporate IAF (Kingma et al., 2016), which is a special variant of normalizing flows (Rezende & Mohamed, 2015). IAF transforms a random noise from a simple distribution into a complex data distribution via parallel sampling from the latent variables.

However, training IAF parameters via a maximum likelihood estimation requires sequential computations for each time step, which is computationally demanding and impractical. Instead, parallel speech synthesis models utilize a *probability density distillation* training strategy (Van Den Oord et al., 2017) to circumvent the issue, which enables the parallel IAF training. The strategy leverages a pre-trained WaveNet as a teacher and a student IAF is trained to minimize the KL divergence between the estimation from the student IAF and the teacher WaveNet, given conditions (e.g., mel spectrogram, linguistic features). The student IAF is optimized such that it can perform the density estimation by closely following the well-trained autoregressive WaveNet.

Parallel WaveNet (Van Den Oord et al., 2017) uses MoL distribution for the pre-trained teacher WaveNet, which does not have a closed-form solution for the KL divergence. The model instead approximates the KL divergence via Monte Carlo estimation, which employs a trade-off between stability and speed, in which the model should generate enough

[1] https://github.com/ksw0306/FloWaveNet

[2] https://github.com/ksw0306/ClariNet

[3] https://ksw0306.github.io/flowavenet-demo

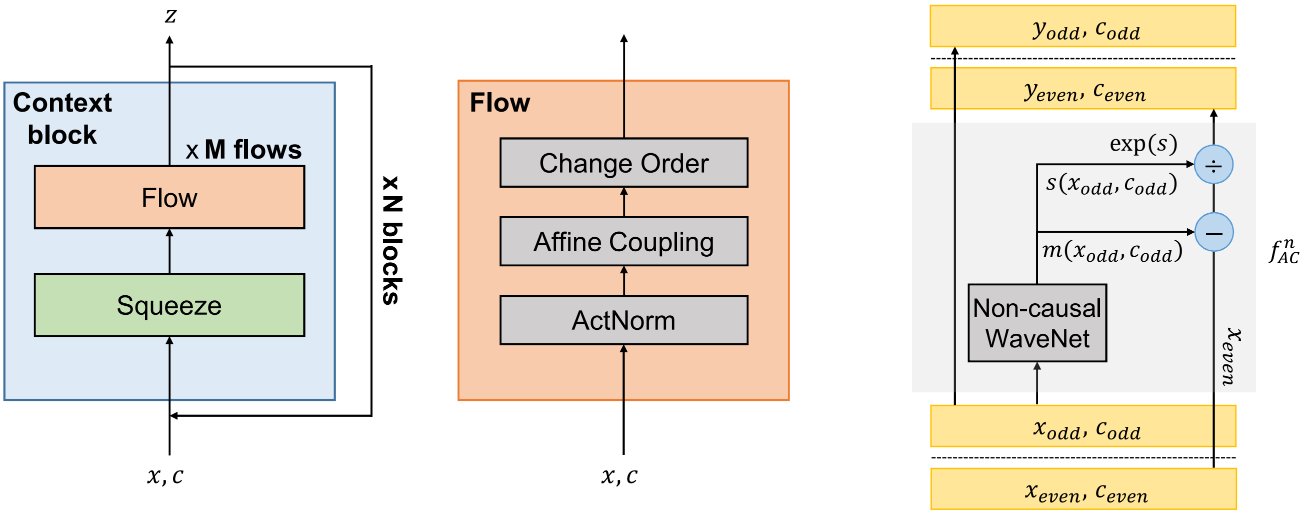

Figure 1. Schematic diagram of FloWaveNet. Left: an entire forward pass of the FloWaveNet consisting of N context blocks. Middle: an abstract diagram of the flow operation. Right: a detailed version of the affine coupling operation.

samples from IAF to ensure stabilized training.

ClariNet (Ping et al., 2018) suggested an alternative single Gaussian formulation for the KL divergence, which is closed-form and mitigates the instability of the Monte Carlo approximation of Parallel WaveNet. ClariNet additionally regularized the KL divergence to compensate for numerical stability when the difference in the standard deviation between a high peak teacher distribution $P_T = \mathcal{N}(\mu_T, \sigma_T^2)$ and the $P_S = \mathcal{N}(\mu_S, \sigma_S^2)$ from the student IAF becomes large, as follows:

$$KL(P_S||P_T) = \log\frac{\sigma_S}{\sigma_T} + \frac{\sigma_S^2 - \sigma_T^2 + (\mu_T - \mu_S)^2}{2\sigma_T^2}, \quad (2)$$

$$KL_{reg}(P_S||P_T) = KL(P_S||P_T) + \lambda|\log\sigma_T - \log\sigma_S|^2. \quad (3)$$

However, both of the aforementioned approaches for parallel speech synthesis require heavily-engineered auxiliary losses for the stabilized training because if the student IAF model uses only probability density distillation, it collapses to a certain mode of the teacher WaveNet distribution. Parallel WaveNet and ClariNet both used an additional spectral distance loss between the synthesized and original audio to ensure realistic sound quality. In addition, Parallel WaveNet further employed extra losses for additional improvements of the model, such as perceptual loss and contrastive loss.

3. FloWaveNet

Here, we describe FloWaveNet, which learns to maximize the exact likelihood of the data while maintaining the ability of real-time parallel sampling, as an alternative to the two-stage training from the related work. FloWaveNet is a hierarchical architecture composed of context blocks as a highest abstract module and multiple reversible transformations inside the context block, as illustrated in Figure 1.

3.1. Flow-based generative model

FloWaveNet is a flow-based generative model using normalizing flows (Rezende & Mohamed, 2015) to model raw audio data. Given a waveform audio signal x, assume that there is an invertible transformation $f(x) : x \longrightarrow z$ that directly maps the signal into a known prior P_Z. We can explicitly calculate the log probability distribution of x from the prior P_Z by using a change of variables formula as follows:

$$\log P_X(x) = \log P_Z(f(x)) + \log\det(\frac{\partial f(x)}{\partial x}). \quad (4)$$

The flow-based generative model can realize the efficient training and sampling by fulfilling the following properties: (i) calculation of the Jacobian determinant of the transformation f should be tractable in equation (4), (ii) mapping random noise z into audio sample x by applying the inverse transformation $x = f^{-1}(z)$ should be efficient enough to compute. Note that the parallel sampling becomes computationally tractable only if property (ii) is satisfied.

To construct a parametric invertible transformation f that fulfills both properties, FloWaveNet uses affine coupling layers suggested in real NVP (Dinh et al., 2016). To model the data using a transformation f that is complex and flexible enough for audio, FloWaveNet stacks multiple flow operations inside each block, comprising the WaveNet affine coupling layers f_{AC} and activation normalization f_{AN} as in Figure 1. The log determinant of the transformation f in Equation (4) can be decomposed into the sum of per-flow terms as follows:

$$\log\det(\frac{\partial f(x)}{\partial x}) = \sum_{n=1}^{MN} \log\det(\frac{\partial(f_{AC}^{n} \cdot f_{AN}^{n})(x)}{\partial x}), \quad (5)$$

where N and M are the number of blocks and flows, respectively.

The change of variables formula in Equation (4) holds for a conditional distribution as well. FloWaveNet can estimate the conditional probability density by incorporating any arbitrary context as the conditional information. In this study, we use the mel spectrogram as a local condition c for the network to model the conditional probability $p(x|c)$, similar to WaveNet used as a vocoder in a common neural Text-to-Speech pipeline (Shen et al., 2018).

3.2. Affine Coupling Layer

A typical flow-based neural density estimator focuses solely on the density estimation as its main objective. The neural density estimator family has the advantage of using a much more flexible transformation, such as masked autoregressive flow (Papamakarios et al., 2017) and transformation autoregressive network (Oliva et al., 2018). However, it only satisfies property (i) and not (ii), making it unusable for our purpose.

In contrast, the affine coupling layer is a parametric layer suggested in real NVP (Dinh et al., 2016) that satisfies both (i) and (ii), and can sample x from z efficiently in parallel. The layer enables the efficient bidirectional transformation of f by making the transformation function bijective while maintaining computational tractability. Each layer is the parametric transformation $f_{AC}^{n} : x \longrightarrow y$, which keeps half of the channel dimension identical and applies the affine transformation only on the remaining half, as follows:

$$y_{odd} = x_{odd}, \quad (6)$$

$$y_{even} = \frac{x_{even} - m(x_{odd}, c_{odd})}{\exp(s(x_{odd}, c_{odd}))}, \quad (7)$$

where m and s are a shared non-causal WaveNet architecture, and c is the local condition (e.g., mel spectrogram) fed by WaveNet to model the conditional probability $p(x|c)$.

Similarly, for the inverse transformation $(f_{AC}^{n})^{-1} : y \longrightarrow x$, we have:

$$x_{odd} = y_{odd}, \quad (8)$$

$$x_{even} = y_{even} \odot \exp s(y_{odd}, c_{odd}) + m(y_{odd}, c_{odd}). \quad (9)$$

Note that the forward and inverse transformations use the same architecture m and s, thus the model satisfies property (ii), which endows the system with the ability of efficient sampling from f^{-1}. The Jacobian matrix is lower triangular, and the determinant is a product of the diagonal elements, which satisfies property (i):

$$\log\det(\frac{\partial f_{AC}^{n}(x)}{\partial x}) = -\sum_{even} s(x_{odd}, c_{odd}). \quad (10)$$

A single affine coupling layer does not alter half of the feature, keeping it identical. To construct a more flexible transformation f, FloWaveNet stacks multiple flow operations for each context block. After the affine coupling of each flow, the change order operation swaps the order of y_{odd} and y_{even} before feeding them to the next flow so that all channels can affect each other during subsequent flow operations.

3.3. Context Block

The context block is the highest abstraction module of FloWaveNet. Each context block consists of a squeeze operation followed by stacks of flow. The squeeze operation takes the data x and condition c, then doubles the channel dimension C by splitting the time dimension T in half, as illustrated in Figure 2. This operation doubles the effective receptive field per block for the WaveNet-based flow, which is conceptually similar to the dilated convolutions of the WaveNet itself. By applying the squeeze operation at the beginning of each context block, the upper-level blocks can have the potential to learn the long-term characteristics of audio, while the lower-level blocks can focus on high-frequency information. The flow operation inside the block contains an activation normalization, affine coupling layer, and change order operation, as described above.

We employed a multi-scale architecture suggested in real NVP (Dinh et al., 2016). The multi-scale model factors out half of the feature channels, to be modeled as a Gaussian earlier after the pre-defined set of several context blocks and the remaining feature channels undergo subsequent context blocks. Thus, for the flows in the higher context blocks, the factored out channels act as identity mapping, whereas the remaining channels are further transformed. We used the same WaveNet-like architecture to estimate the mean and variance of the factored out Gaussian, using the remaining feature channels as inputs. This estimation strategy has minimal impact on the speed of synthesis compared to the

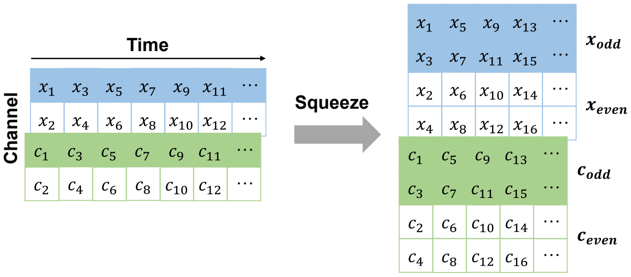
Time
Channel
x_1 x_3 x_5 x_7 x_9 x_{11} $\cdots$
x_2 x_4 x_6 x_8 x_{10} x_{12} $\cdots$
c_1 c_3 c_5 c_7 c_9 c_{11} $\cdots$
c_2 c_4 c_6 c_8 c_{10} c_{12} $\cdots$
Squeeze
x_1 x_5 x_9 x_{13} $\cdots$
x_3 x_7 x_{11} x_{15} $\cdots$
x_{odd}
x_2 x_6 x_{10} x_{14} $\cdots$
x_4 x_8 x_{12} x_{16} $\cdots$
x_{even}
c_1 c_5 c_9 c_{13} $\cdots$
c_3 c_7 c_{11} c_{15} $\cdots$
c_{odd}
c_2 c_6 c_{10} c_{14} $\cdots$
c_4 c_8 c_{12} c_{16} $\cdots$
c_{even}

Figure 2. Squeeze operation used in the context block.

pre-defined mean and variance of the factored out Gaussian, while producing higher-quality audio.

3.4. Activation Normalization

The activation normalization (ActNorm) layer suggested in Glow (Kingma & Dhariwal, 2018) stabilizes the training of the network composed of multiple flow operations. The ActNorm layer f_{AN}^n is a per-channel parametric affine transformation at the beginning of the flow. For i-th channel, we have:

$$f_{AN}^n(x_i) = x_i * s_i + b_i, \tag{11}$$

where s and b represent scale and bias for each channel.

Note that f_{AN}^n is a volume-changing operation, thus the log-determinant of the operation is computed as follows:

$$\log\det(\frac{\partial f_{AN}^n(x)}{\partial x}) = T * \sum_{i=1}^{C} \log|s^i|. \tag{12}$$

The layer performs data-dependent initialization of the trainable parameters s and b by scaling the activation channel-wise to have zero mean and unit variance for the first given batch of data.

4. Experiments

We trained the model using the LJSpeech dataset (Ito, 2017), which is a 24-hour waveform audio set of a single female speaker with 13,100 audio clips and a sample rate of 22kHz. We randomly extracted 16,000 sample chunks and normalized them to $[-1, 1]$ as the input. For local conditioning with the mel spectrogram construction, we used a preprocessing method from Tacotron 2 (Shen et al., 2018). The generated 80-band mel spectrogram is used by the network to estimate the conditional probability.

We reproduced results from the original autoregressive WaveNet (Van Den Oord et al., 2016) and the Gaussian IAF of ClariNet (Ping et al., 2018) as baselines. All models are trained under the mel spectrogram condition. We used an Adam optimizer (Kingma & Ba, 2014) with a learning rate of 10^{-3} for all models, identically to the ClariNet training configuration. We scheduled the learning rate decay by a factor of 0.5 for every 200K iterations. We used NVIDIA Tesla V100 GPUs with a batch size of 8 for all models.

4.1. Autoregressive WaveNet

We trained two autoregressive WaveNet models, one with the MoL and the other with a single Gaussian, as the output distribution. For the MoL WaveNet, we trained the best-performing autoregressive WaveNet from Tacotron 2 (Shen et al., 2018), with the exact configuration from the paper, which is a 24-layer architecture with four 6-layer dilation cycles. For the single Gaussian WaveNet, we trained the 20-layer model with the same configuration used in ClariNet (Ping et al., 2018). We trained the models for a total of 1M iterations.

4.2. Gaussian Inverse Autoregressive Flow (IAF)

For the Gaussian IAF from ClariNet, we used the best-performing pre-trained single Gaussian autoregressive WaveNet from subsection 4.1 as the teacher network for the probability density distillation. We used the transposed convolution parameters from the teacher network without further tuning to upsample the mel spectrogram condition.

The student network has an architecture similar to the teacher network. It has a 60-layer architecture with six stacks of IAF modules, each of them with a 10-layer dilation cycle, which is the same configuration that is used in ClariNet (Ping et al., 2018). We trained the model for a total of 500K iterations.

The ClariNet training requires a regularized KL divergence loss and an auxiliary spectrogram frame loss. In addition to the standard training, we performed an analysis of the impact of each loss by additionally training models with only one of the two losses.

4.3. FloWaveNet

FloWaveNet has 8 context blocks. Each block contains 6 flows, which results in a total of 48 stacks of flows. We used the affine coupling layer with a 2-layer non-causal WaveNet architecture (Van Den Oord et al., 2016) and a kernel size of 3 for each flow. We used the multi-scale architecture described in 3.3, in which we factored out half of the feature channels as a Gaussian after 4 context blocks, and we estimated the mean and variance of the Gaussian using the 2-layer WaveNet architecture, identically to the affine coupling layer.

Table 1. Comparative mean opinion score (MOS) results with 95% confidence intervals and conditional log-likelihoods (CLL) on test set.

METHODS	5-SCALE MOS	TEST CLL
GROUND TRUTH	4.67± 0.076	
MOL WAVENET	4.30± 0.110	4.6546
GAUSSIAN WAVENET	4.46± 0.100	4.6526
GAUSSIAN IAF	3.75± 0.159	
FLOWAVENET	3.95± 0.154	4.5457

Table 2. Training iterations per second and inference speed comparison. The values for WaveNet and Parallel WaveNet are from (Van Den Oord et al., 2017), and the others are from our implementation.

METHODS	ITER/SEC	SAMPLES/SEC
WAVENET	N/A	172
PARALLEL WAVENET	N/A	500K
GAUSSIAN WAVENET	1.329	44
GAUSSIAN IAF	0.636	470K
FLOWAVENET	0.714	420K

We used 256 channels for a residual, skip, and gate channel with a gated tanh activation unit for all of the WaveNet architecture, along with the mel spectrogram condition. The weights for the last convolutional layer of the affine coupling are initialized with zero, so that it simulates identity mapping during the initial stage of training, which reportedly showed a stable training result. (Kingma & Dhariwal, 2018)

We induced the model to learn the long-term dependency of audio by stacking many context blocks, instead of increasing the dilation cycle of the affine coupling. We trained the model with a single maximum likelihood loss without any auxiliary terms for 700K iterations.

5. Results and Analysis

Our results with the sampled audio for all models are publicly available as described in Section 1.

The Gaussian IAF and FloWaveNet generate the waveform audio by applying the normalizing flows, using random noise samples as inputs. We set the standard deviation (*i.e.*, temperature) of the prior below 1, which generated relatively higher-quality audio. We chose a temperature of 0.8 for FloWaveNet as the default, which empirically showed the best sound quality.

5.1. Model Comparisons

Table 1 presents comparative results from a subjective 5-scale MOS experiment via Amazon Mechanical Turk, along with an objective conditional log-likelihoods (CLL) on test set. In general, the autoregressive WaveNet (MoL or Gaussian) received the highest CLL and MOS, which are closest to the ground truth sample. Between the autoregressive models, the Gaussian WaveNet performed slightly better than the MoL WaveNet, which is consistent with the result from ClariNet (Ping et al., 2018).

The parallel speech synthesis models showed a slightly degraded audio quality compared to the autoregressive model, as also objectively evidenced by the lower test CLL. FloWaveNet showed a better performance evaluation result compared to that of our reproduced version of the Gaussian IAF. For the Gaussian IAF, there was an audible white noise throughout the samples which might negatively affect the evaluation of the audio quality compared to other models. FloWaveNet did not incur the white noise and had a clear sound quality unlike the Gaussian IAF. But instead, there was a periodic artifact perceived as a trembling voice, which varied for different temperatures and we discuss it in Section 5.2.

Overall, the sound quality of FloWaveNet was comparable to the previous approaches as well as having the advantages of the single maximum likelihood loss and the single-stage training. Note that although the autoregressive WaveNet showed the highest fidelity, it requires slow ancestral sampling. Table 2 shows the inference speed comparison for each model. FloWaveNet can generate the 22,050Hz audio signal approximately 20 times faster than real-time, which is similar in magnitude to the reported speed results from the Parallel WaveNet (Van Den Oord et al., 2017).

Our reimplemented Gaussian IAF from the ClariNet model achieved a sampling speed approximately similar to FloWaveNet. Note that the difference in sampling speed between various parallel speech synthesis architectures (including our FloWaveNet model) is largely from a selection of the number of channels of convolutional layers. The Parallel WaveNet and ClariNet used 64 channels for the reported model architecture in their respective papers. The experimental results from FloWaveNet used 256 channels as our default settings. However, one can also construct a smaller version of FloWaveNet with 128 channels which results in a slightly degraded performance but with twice the sampling speed.

We also reported the number of training iterations per second for Gaussian WaveNet, Gaussian IAF, and FloWaveNet with a single NVIDIA Tesla V100 GPU in Table 2. Note that we chose this metric as the most relevant and dataset-independent speed benchmark for training. As the Gaussian IAF requires two-stage training procedure, its total training wall-clock time is the sum of training Gaussian WaveNet and Gaussian IAF: Each of them takes 8.7 and 9.1 GPU days for convergence on the LJSpeech with the aforementioned

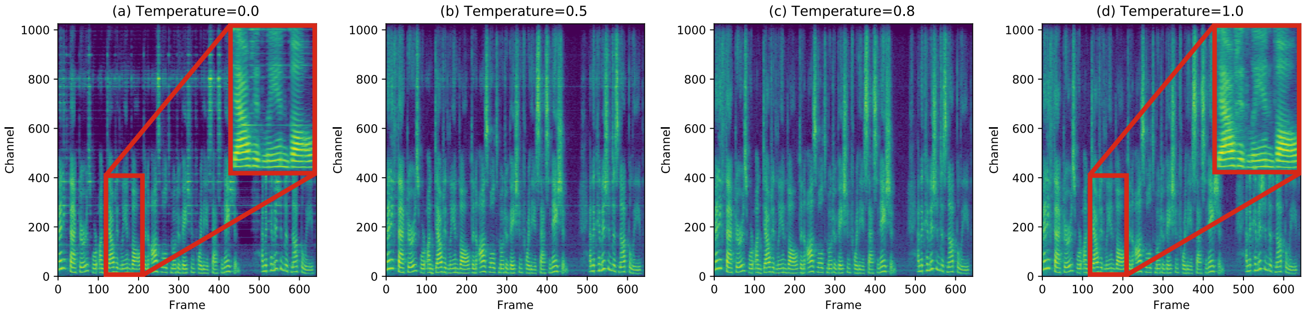

Figure 3. Spectrogram visualization of FloWaveNet samples with different temperatures.

training iterations, respectively. Training FloWaveNet takes 11.3 GPU days, demonstrating that FloWaveNet has a practical merit of faster convergence since it does not necessitate the well-trained teacher network.

5.2. Temperature Effect on Audio Quality Trade-off

The empirical experiments presented in Glow (Kingma & Dhariwal, 2018) included adjusting the temperature of the pre-defined known prior. Glow maximized the perceptive quality of generated images by drawing random samples with the temperature below 1 and then applying inverse transformation to the random samples. We performed a similar empirical study on the effects of the temperature in terms of audio.

The flow-based generative models can perform an interpolation between two data points in the latent space via the latent variables. We interpret the temperature effect from the perspective of a latent traversal, where lowering the temperature T corresponds to performing the latent traversal between a random sample z from the known prior and a zero vector O:

$$P_{\hat{Z}}(\hat{z}) = \mathcal{N}(0, T^2) \iff \hat{z} = (1 - T) * O + T * z. \quad (13)$$

Figure 3 represents the visualized spectrogram of the latent traversal with temperatures of 0, 0.5, 0.8, and 1.0. The spectrogram in Figure 3 (a) corresponds to the audio using the zero vector. We can see that it exhibits many horizontal lines, which correspond to constant noises for multiple pitches, while instead, the zero vector tends to model the harmonic structure of the speech more strongly with high resolution, as can be seen in the zoomed-in view. In contrast, the spectrogram in Figure 3 (d) shows that the temperature of 1.0 generates a high-quality audio without the constant noise artifacts. However, it is harder to capture the harmonic structure of the speech content using the high temperature, which is translated into the trembling voice.

Table 3. KL divergence results on test data, estimated between each Gaussian IAF and the teacher WaveNet.

METHOD: GAUSSIAN IAF	KL DIVERGENCE
ONLY KL	0.040
KL + FRAME	0.134
ONLY FRAME	1.378

From the latent traversal analysis, we can see that there exists a trade-off between high-fidelity quality and the harmonic structure of the speech. We could generate high-quality speech via lowering the temperature to minimize the trembling, while also ensuring that the constant noise artifacts are inaudible and not discomforting. We maximized the perceptive audio quality through the empirical post-processing approach of temperature optimization in this work, and further research on improving flow-based generative architectures would be a promising direction for future work.

5.3. Analysis of ClariNet Loss Terms

Parallel WaveNet (Van Den Oord et al., 2017) and ClariNet (Ping et al., 2018) are parallel waveform synthesis models based on IAF, as discussed earlier. They use KL divergence in tandem with additional auxiliary loss terms to maximize the quality of the sampled audio. Here, we present an empirical analysis of the role of each term. We decomposed the two losses of ClariNet and trained separate Gaussian IAF models with only one of the losses: the KL divergence or the spectrogram frame loss.

Table 3 shows a quantitative analysis on the KL divergence between the estimation from each Gaussian IAF training method and the pre-trained teacher WaveNet, given the test data mel spectrogram condition. Figure 4 represents the spectrogram examples generated by each method. The waveform audio samples corresponding to each method are also publicly available at the provided webpage.

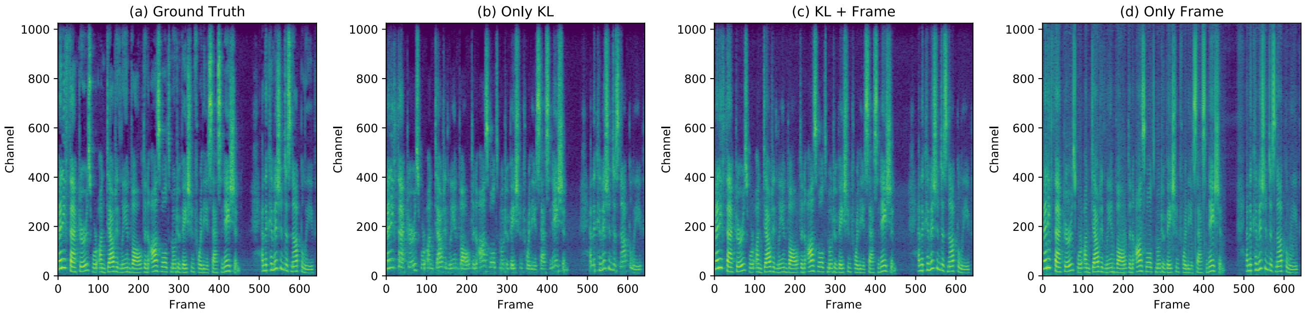

Figure 4. Spectrogram visualization of ClariNet samples with different loss configurations.

Table 4. Mean opinion score (MOS) results with non-causal and causal variants of FloWaveNet.

METHOD: FLOWAVENET	5-SCALE MOS
NON-CAUSAL	3.95 ± 0.154
CAUSAL	3.36 ± 0.134

We can clearly see from Table 3 that for the Gaussian IAF, using only the KL divergence loss showed the best reported metric on the test data, which is a direct result from the explicit optimization. However, samples generated by the KL-only training, although phonetically perceptible, are low-volume and sound distorted. Figure 4 (b) shows that the signal has low-energy across all samples compared to the ground truth and that the model has limitations on estimating the harmonics of the speech signal in the mid-frequency range ($e.g.$, 2∼5kHz). Not only does this show that a good KL divergence result does not necessarily mean a realistic-sounding audio, but also the Gaussian IAF is prone to mode collapse problem when using the probability density distillation loss only.

The model trained with only the spectrogram frame loss showed relatively fast estimation of the acoustic content of the original audio earlier in training ($e.g.$, 10K iterations). The spectrogram in Figure 4 (b) more closely resembles the ground truth spectrogram because the frame loss directly targets the resemblance in the frequency domain. However, the generated samples showed a high amount of noise and the noise did not diminish without the KL divergence loss in the remaining training iterations.

Ideally, the probability density distillation enables the student network perfectly mimic the distribution estimated by the teacher WaveNet. However, the distilled model trained only by the KL divergence shows its limitations of covering every mode of the teacher model, as evidenced by this study and by previous works (Van Den Oord et al., 2017). This shortcoming can be alleviated by incorporating the spectrogram frame loss into the distillation process, as shown by the Figure 4 (c), where the model starts to appropriately track the harmonics and estimate the original amplitude. Thus, the Gaussian IAF training requires both complementary loss terms for realistic-sounding speech synthesis.

5.4. Causality of WaveNet Dilated Convolutions

The original WaveNet achieved autoregressive sequence modeling by introducing causal dilated convolutions. However, for FloWaveNet, the causality is no longer a requirement because the flow-based transformation is inherently parallel in either direction. We performed an ablative study by comparing both approaches, and the non-causal version of FloWaveNet exhibited better sound quality, as can be seen in Table 4. This is because the non-causal version of FloWaveNet has the benefit of observing the mel spectrogram condition both forward and backward in its receptive field.

6. Conclusion

In this paper we proposed FloWaveNet, a flow-based generative model that can achieve a real-time parallel audio synthesis that is comparable in fidelity to two-stage approaches. Thanks to the simplified single loss function and single-stage training, the model can mitigate the need for highly-tuned auxiliary loss terms while maintaining stability of training which is useful in practical applications. Our results show that the flow-based generative model is a promising approach in speech synthesis domain which shed light on new research directions.

Acknowledgements

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (Ministry of Science and ICT) [No. 2018R1A2B3001628], Hyundai Motor Company, AIR Lab (AI Research Lab) in Hyundai Motor Company through HMC-SNU AI Consortium Fund, Samsung Electronics Company, and Brain Korea 21 Plus Project in 2019.

References

Arik, S., Diamos, G., Gibiansky, A., Miller, J., Peng, K., Ping, W., Raiman, J., and Zhou, Y. Deep voice 2: Multi-speaker neural text-to-speech. *arXiv preprint arXiv:1705.08947*, 2017a.

Arik, S. O., Chrzanowski, M., Coates, A., Diamos, G., Gibiansky, A., Kang, Y., Li, X., Miller, J., Ng, A., Raiman, J., et al. Deep voice: Real-time neural text-to-speech. *arXiv preprint arXiv:1702.07825*, 2017b.

Dinh, L., Sohl-Dickstein, J., and Bengio, S. Density estimation using real nvp. *arXiv preprint arXiv:1605.08803*, 2016.

Ito, K. The lj speech dataset. https://keithito.com/LJ-Speech-Dataset/, 2017.

Jia, Y., Zhang, Y., Weiss, R. J., Wang, Q., Shen, J., Ren, F., Chen, Z., Nguyen, P., Pang, R., Moreno, I. L., et al. Transfer learning from speaker verification to multispeaker text-to-speech synthesis. *arXiv preprint arXiv:1806.04558*, 2018.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Kingma, D. P. and Dhariwal, P. Glow: Generative flow with invertible 1x1 convolutions. *arXiv preprint arXiv:1807.03039*, 2018.

Kingma, D. P., Salimans, T., Jozefowicz, R., Chen, X., Sutskever, I., and Welling, M. Improved variational inference with inverse autoregressive flow. In *Advances in Neural Information Processing Systems*, pp. 4743–4751, 2016.

Oliva, J. B., Dubey, A., Póczos, B., Schneider, J., and Xing, E. P. Transformation autoregressive networks. *arXiv preprint arXiv:1801.09819*, 2018.

Papamakarios, G., Murray, I., and Pavlakou, T. Masked autoregressive flow for density estimation. In *Advances in Neural Information Processing Systems*, pp. 2338–2347, 2017.

Ping, W., Peng, K., Gibiansky, A., Arik, S. O., Kannan, A., Narang, S., Raiman, J., and Miller, J. Deep voice 3: 2000-speaker neural text-to-speech. *arXiv preprint arXiv:1710.07654*, 2017.

Ping, W., Peng, K., and Chen, J. Clarinet: Parallel wave generation in end-to-end text-to-speech. *arXiv preprint arXiv:1807.07281*, 2018.

Rezende, D. J. and Mohamed, S. Variational inference with normalizing flows. *arXiv preprint arXiv:1505.05770*, 2015.

Shen, J., Pang, R., Weiss, R. J., Schuster, M., Jaitly, N., Yang, Z., Chen, Z., Zhang, Y., Wang, Y., Skerrv-Ryan, R., et al. Natural tts synthesis by conditioning wavenet on mel spectrogram predictions. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4779–4783. IEEE, 2018.

Van Den Oord, A., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., Kalchbrenner, N., Senior, A. W., and Kavukcuoglu, K. Wavenet: A generative model for raw audio. In *SSW*, pp. 125, 2016.

Van Den Oord, A., Li, Y., Babuschkin, I., Simonyan, K., Vinyals, O., Kavukcuoglu, K., Driessche, G. v. d., Lockhart, E., Cobo, L. C., Stimberg, F., et al. Parallel wavenet: Fast high-fidelity speech synthesis. *arXiv preprint arXiv:1711.10433*, 2017.