Правительство Российской Федерации
Федеральное государственное автономное образовательное
учреждение высшего образования
«Национальный исследовательский университет
«Высшая школа экономики»

Факультет Санкт-Петербургская школа
физико-математических и компьютерных наук
Основная образовательная программа
«Промышленное программирование»

Беляев Владислав Валерьевич

Улучшение синтеза голоса с участием генеративно-состязательных сетей

Магистерская диссертация

Допущен к защите.
Зав. кафедрой:
д. ф.-м. н., профессор Омельченко А. В.

Научный руководитель:
Кураленок И. Е.

Рецензент:
Борис Гинсбург

Санкт-Петербург
2021

SAINT-PETERSBURG ACADEMIC UNIVERSITY
Higher education centre

Department of Mathematics and Information Technology

Belyaev Vladislav

Improving Voice Synthesis Involving Generative Adversarial Networks

Graduation Thesis

Admitted for defence.
Head of the chair:
professor Alexander Omelchenko

Scientific supervisor:
Igor Kuralenok

Reviewer:
Boris Ginsburg

Saint-Petersburg
2021

Оглавление

Введение **5**

1. Обзор предметной области **7**

2. Аналоги **9**

2.1. Методы обработки чистого сигнала 9

2.2. Модели на основе авторегрессионных нейронных сетей . 10

2.3. Неавторегрессионные модели 11

2.4. Модели на основе генеративно-состязательных сетей . . 13

2.5. Особенности предыдущих решений 16

3. Методы **20**

3.1. Генератор 20

3.1.1. Общая архитектура 20

3.1.2. Cascade Block 21

3.1.3. Замена Deconv блоков 23

3.1.4. Функция потерь 24

3.2. Дискриминатор 27

3.2.1. Типы дискриминаторов 27

3.2.2. Обуславливание на мел-спектрограмме 27

3.2.3. U/V коррекция 28

3.2.4. Функции потерь 29

3.3. Общая структура обучения модели 30

3.4. Unsupervised до-обучение модели 31

4. Эксперименты **33**

4.1. Описание экспериментов 33

4.1.1. Используемые метрки 33

4.1.2. Используемые датасеты 34

4.1.3. Особенности обучения моделей 35

4.2. Результаты экпериментов 35

4.3. Результаты экспериментов до-обучения на мел-спектрограммах с ошибкой 36

Выводы **37**

Список литературы **39**

Введение

Голос - это один из наиболее часто используемых и естественных коммуникационных интерфейсов для людей. Последние достижения в области синтеза речи в подавляющем большинстве случаев связаны с глубоким обучением или сквозными методами, которые используются для широкого спектра сценариев применения, таких как интеллектуальное речевое взаимодействие, умные дома, чатботы или разговорный искусственный интеллект.

Моделирование звука представляет собой особенно сложную задачу из-за высокого временного разрешения данных и наличия структур на различных временных масштабах с краткосрочными и долгосрочными зависимостями. Большинство современных подходов используют мел-спектрограммы как промежуточное низкоразмерное представление в задачах Text-To-Speech или Voice Conversion [18] [45]. Задача моделей синтеза звука (вокодеров) эффективно инвертировать мел-спектрограмму в аудио.

Вокодеры основанные на генеративно состязательных сетях (GAN) [16] делают значительные прорывы в этой области в последние годы, превосходя авторегрессионые и Flow-подобные аналоги по характеристикам качества и скорости работы [20]. Однако широкое применение GAN-подобных вокодеров на реальных задачах может быть ограничего проблемой устойчивости к ошибке мел-спектрограммы на входе для сложных целевых задач, например таких как задача замены голоса (Voice Conversion). Поэтому очень важно уметь правильно адаптировать вокодер к подобной ошибке. Цель данной работы: разработать вокодер, основанный на генеративно состязательных сетях, улучшив качество голоса и устойчивость к зашумленным мелам.

В работе предложена новая архитектура CascadeGAN, улучшающего как структуру генератора, так и структуру дискриминатора относительно базового MelGAN [25]. Модель превосходит бейзланы по метрикам качества, повышая PESQ на 0.06 на датасете LibriTTS.

Также в работе разработан и исследован новый unsupervised спо-

соб адаптации моделей к ошибке на входе, не требующий правильной аудио разметки и поэтому являющийся общим для широкого класса задач. Подобное до-обучение для модели CascadeGAN на целевой задаче замены голосы повысило метрику MOS на 0.49, убрав значительное число аудио артефактов и шумов.

1. Обзор предметной области

В последнее время с развитием нейронных сетей технология синтеза речи стала стремительно развиваться. Большинство моделей нейронного синтеза речи используют двухэтапный процесс: 1) предсказание промежуточного представления аудио с низким разрешением, например какой-нибудь низко-уровневой акустической характеристики и 2) синтез необработанного звукового сигнала из промежуточного представления. Первый этап - это модели класса Text-to-Speech (TTS) или Voice-Conversion (VC), на выходе которых получаются промежуточные представления аудио, тогда как модели второго этапа синтезирует необработанные аудио сигналы с частотой до 44000 измерений в секунду, и должны учитывать как краткосрочные, так и более широкомасштабные зависимости в аудио волне. В этой работе мы сосредоточены на разработке модели второго этапа (вокодера), которая эффективно синтезирует аудио из низкоразмерного представления.

Мел-частотная спектрограмма связана с линейно-частотной спектрограммой, т.е. величиной кратковременного преобразования Фурье (STFT). Она получается путем применения нелинейного преобразования к частоте оси STFT, основана на статистической обработке большого числа данных о субъективном восприятии высоты звуковых тонов и суммирует частотное представление с меньшим количеством измерений. Мел-спектрограмма призвана имитировать нелинейное восприятие звука человеческим ухом, будучи более разборчивой на низких частотах и менее разборчивой на высоких. Использование такой слуховой шкалы частот имеет следующий эффект: модель делает акцент на деталях на низких частотах, которые имеют решающее значение для разборчивости речи, в то время как на высоких частотах, где доминируют фрикативные и другие шумовые всплески, как правило, не требуется высокой точности моделирования. Такое представление также более гладкое, чем аудио волна, и его легче обучить с помощью среднеквадратичной ошибки поскольку, Также оно инвариантно к фазе в пределах каждой мел-позиции. Благодаря этим свойствам,

характеристики, полученные из мел-спектрограмм, использовались в качестве базовых для распознавания речи на протяжении многих десятилетий [7].

2. Аналоги

Современные подходы к инверсии мел-спектограмм можно разделить на категории на три отдельных семейства: методы обработки чистого сигнала, модели с авторегрессией и неавторегрессионные нейронные сети. Мы опишем эти три основных направления исследований в следующих параграфах. Затем мы изложим принцип действия неавторегрессионных моделей нового типа, основанных на генеративно-состязательных сетях[16] и рассмотрим некоторые особенности последних при использовании на целевой задаче замены голоса.

2.1. Методы обработки чистого сигнала

Были изучены различные подходы к обработке сигналов, чтобы найти удобные представления аудио с низким разрешением, которые можно легко моделировать и эффективно преобразовывать обратно в аудиосигнал во временной области. Например, алгоритм Griffin-Lim [13] позволяет эффективно декодировать последовательность STFT обратно во временной сигнал ценой появления сильных роботизированных аудио артефактов, как отмечено в [41]. Были также исследованы более сложные представления и методы обработки чистого сигнала. Например, вокодер WORLD [29] вводит промежуточное представление, предназначенное для моделирования речи на основе мел-спектрограммы. WORLD работает в паре со специальным алгоритмом обработки сигнала для преобразования промежуточного представления в исходный звук. Он был успешно использован для задачи Text-To-Speech (TTS), например, в модели Char2Wav, где функции вокодера WORLD моделируются с помощью рекуррентной нейронной сети на основе механизма внимания [37] [36] [33]. Несмотря на относительно высокое быстродействие, основная проблема классических методов обработки сигнала заключается в том, что отображение промежуточных характеристик в аудио волну обычно вносит заметные шумы и неприятные для слуха артефакты.

2.2. Модели на основе авторегрессионных нейронных сетей

WaveNet [2] - это полностью конволюционная авторегрессионная модель, которая способна генерировать высокореалистичные образцы аудио, обуславливаясь на различные лингвистические характеристики, согласованные по времени с исходным аудио. Она также способна генерировать высококачественные безусловные образцы речи и музыки. Мы рассмотрим ее в качестве бейзлайна в данной работе и далее кратко опишем архитектуру. Сама модель состоит из последовательности расширенных сверток с экспоненциально увеличивающимися факторами для увеличения рецептивного поля модели. На рисунке 1 изображеная принципиальная схема WaveNet.

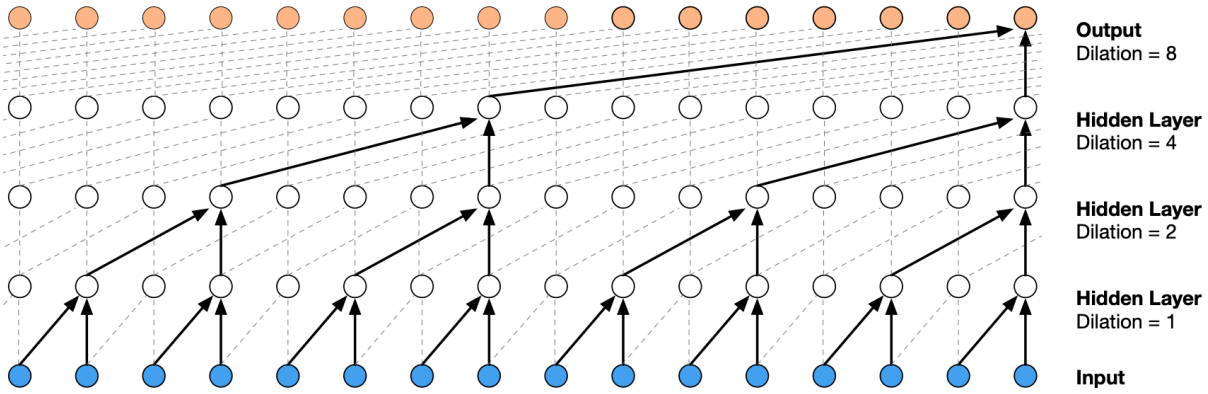

Рис. 1: Визуализация последовательности расширенных сверток, используемых в модели WaveNet. Картинка взята из [2].

Модель является полностью вероятностной и авторегрессионной, с предсказанием распределения для каждого аудио сигнала, обусловленным всеми предыдущими. Тем не менее, в работе [2] показано, что она может быть эффективно обучена на данных с десятками тысяч аудио измерений в секунду. При применении к преобразованию текста в речь (TTS) она значительно улучшает производительность [18], причем ассесоры оценивают ее как значительно более естественно звучащую, чем лучшие параметрические модели для английского и китайского языков. Одна сеть WaveNet может одинаково точно передавать характеристики многих различных дикторов и переключаться

между ними, ориентируясь на личность диктора. Однако для готовых к внедрению на реальные задачи систем генерации речи также важна генерация в реальном времени на GPU и CPU. Модель WaveNet является чрезвычайно медленной из-за последовательной авторегрессионной структуры, и не может быть эффективно использована для широкого класса применений синтеза звука,

Были также исследованы и другие авторегрессионные модели. SampleRNN [28] - это альтернативная архитектура для безусловной генерации волновых форм, которая моделирует звук явно в различных временных разрешениях с помощью многомасштабных рекуррентных нейронных сетей. WaveRNN [22] - это более быстрая авторегрессионная модель, основанная на простой однослойной рекуррентной нейронной сети. WaveRNN использует различные методы, такие как спарсификация и генерация подмасштабов для дальнейшего увеличения скорости синтеза. Эти методы позволили добиться передовых результатов в синтезе текста в речь [37] [36] [33] и других задачах генерации звука [9]. К сожалению, применение с помощью этих моделей по своей природе также является медленным и неэффективным, поскольку аудио измерения должны генерироваться последовательно.

2.3. Неавторегрессионные модели

Значительные усилия были направлены на разработку моделей без авторегрессии для инвертирования аудиопредставлений низкого разрешения. Эти модели на порядки быстрее своих авторегрессионных аналогов, поскольку они хорошо распараллеливаются и могут полностью использовать современные аппаратные средства глубокого обучения (такие как GPU и TPU). Для обучения таких моделей были разработаны два различных метода. 1.) Parallel Wavenet [32] и Clarinet [10] дистиллируют обученный авторегрессивный вокодер в потоковую конволюционную модель студента. Студент обучается с помощью задачи дистилляции вероятностей, основанной на расхождении Кульбека-Лейблера: $KL[P_{\text{student}}||P_{\text{teacher}}]$, а также дополнительном условии поте-

ри восприятия (perceptual loss). 2.) WaveGlow [11] - это потоковая генеративная модель, основанная на Glow [24], которую мы рассмотрим в качестве бейзлайна в данной работе и кратко опишем в следующих параграфах.

WaveGlow - это генеративный поток с очень высокой пропускной способностью, состоящий из 12 сцепленных (coupling) и 12 инвертируемых сверток 1x1, причем каждый сцепленный слой состоит из стека из 8 слоев расширенных сверток. Благодаря биективности каждого слоя в модели, с помощью WaveGlow можно оптимизировать правдоподобие данных напрямую и избежать тяжело обучаемого дистилирования модели учителя. Визуальная схема модели представлена на 2.

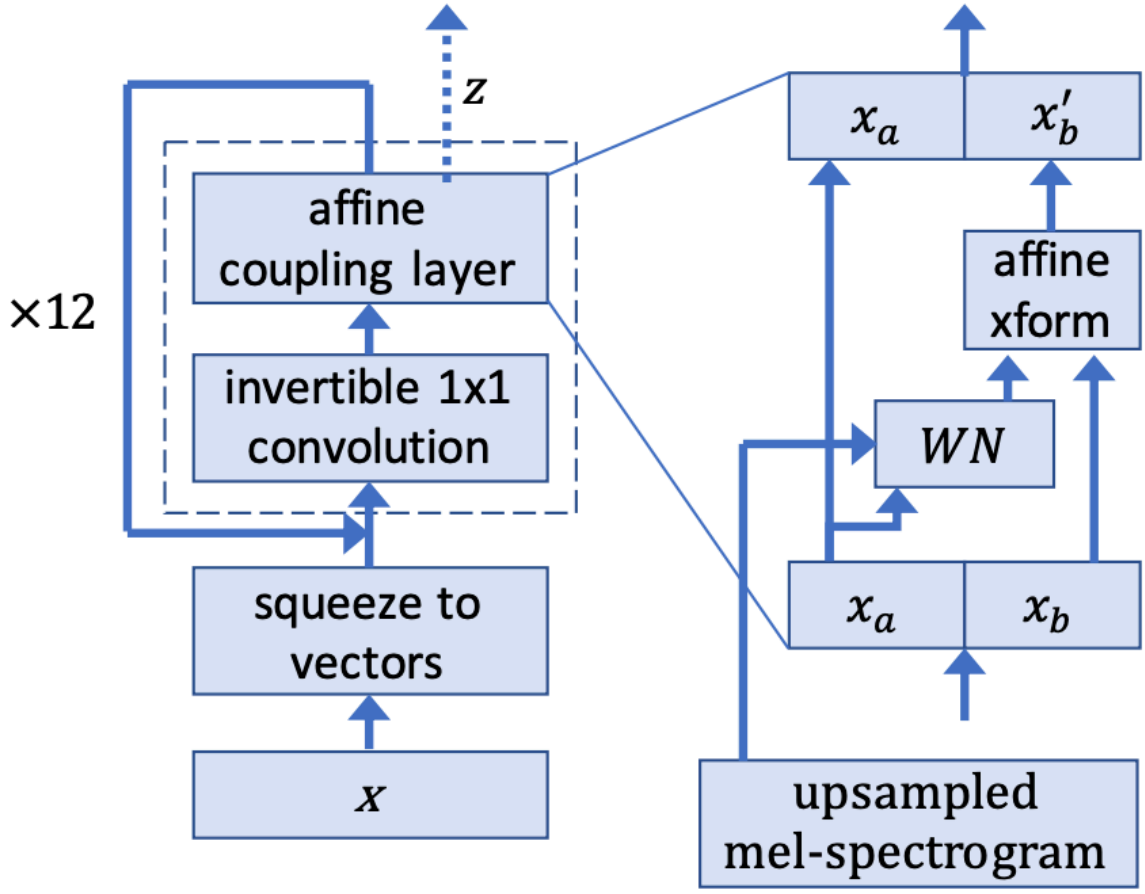

Рис. 2: Архитектура сети WaveGlow. Картинка взята из [11]

WaveGlow сравним с моделью WaveNet [2] по качеству генерации и превосходит по скорости применения на несколько порядков. Авторы отмечают, что для получения качественных результатов для синте-

за речи одного спикера требуется неделя обучения на 8 графических процессорах, что является препятствием для дальнейшего активного исследования модели. Хотя выводы на GPU выполняются быстро, большой размер модели делает ее непрактичной для приложений с ограниченным бюджетом памяти.

2.4. Модели на основе генеративно-состязательных сетей

Одним из семейств методов, которые до сих пор мало изучены для глубокого обучения для звука, являются генеративные состязательные сети (GAN) [16]. GAN'ы продолжают дабиваться устойчивого прогресса в безусловной генерации изображений [14] [23], переводе изображений в изображения [17] [44] [40] и синтезе видео в видео [6] [39]. Мы наблюдаем столь же значительный прогресс в использовании GAN'ов для моделирования звука. [8] используют GAN'ы для генерирования музыкального тембра путем моделирования величин и фазовых углов STFT вместо прямого моделирования необработанной звуковой волны. [30] предлагают использовать GAN'ы для обучения отображения мел-спектрограммы в простую спектрограмму, которая должна быть объединена с оценками фазы для восстановления конечной формы звукового сигнала. [43] используют GAN'ы для дистилляции авторегрессионной модели, которая генерирует аудио. В этих работах было показано, что в качестве важного компонента обучения звуковой модели требуется дистилляция, основанная на KL-дивергенции. Напротив, результаты [25] [35] показали, что adversarial функций потерь достаточно для высококачественной генерации аудиосигнала. Это открыло дорогу стремительному развитию легковесных GAN-подобных архитектур для синтеза звука [20].

В качестве одного из GAN-подобного бейзлайна мы рассмотрим MelGAN [25]. На рисунке 3 изображена архитектура сети, которая состоит из генератора, отображающего мел-спектрограмму в аудио волну и дискриминатора, который учится различать реальное аудио

записи от сгенерированных текущим состоянием генератора. Генератор представляет собой полностью сверточную сеть и, поскольку, мел-спектрограмма (используемая для всех экспериментов и в данной работе) имеет в 256 раз меньшее временное разрешение, используется стек транспонированных конволюционных слоев для повышения дискретизации входной спектрограммы. За каждым транспонированным сверточным слоем следует стек остаточных (residual) блоков с расширенными свертками. В отличие от традиционных GAN'ов, генератор не использует вектор глобального шума в качестве входного сигнала. В экспериментах [25] было замечено, что разница в восприятии генерируемых сигналов незначительна, когда в генератор подается дополнительный шум. Это неинтуитивный результат, поскольку инверсия мел-спектрограммы включает в себя отображение "один-ко-многим", поскольку мел-спектрограмма является сжатием аудио с потерями, однако этот вывод согласуется с данными [27] и [17], которые показывают, что шум не важен, если информация об обуславливании очень сильна.

Дискриминатор MelGAN'а состоит из нескольких частей, для анализа аудио на разных масштабах. Каждый отдельный дискриминатор представляет собой марковский оконный дискриминатор [17], состоящий из последовательности стратифицированных конволюционных слоев с большим размером ядра. В то время как стандартный GAN-дискриминатор учится классифицировать распределения целых аудио-последовательностей, оконный дискриминатор учится поддерживать согласованность между патчами, поскольку, было показано, что они захватывают информацию о важной высокочастотной структуре, требуют меньше параметров, работают быстрее и могут быть применены к аудио последовательностям переменной длины. Согласно результатам экспериментов, MelGAN [25] показывает меньшее качество генерации речи, чем модели с авторегрессией и потоками, однако обладает на порядок большим быстродействием и гораздо меньшим колличеством весов.

Были также исследованы и другие GAN-подобные архитектуры.

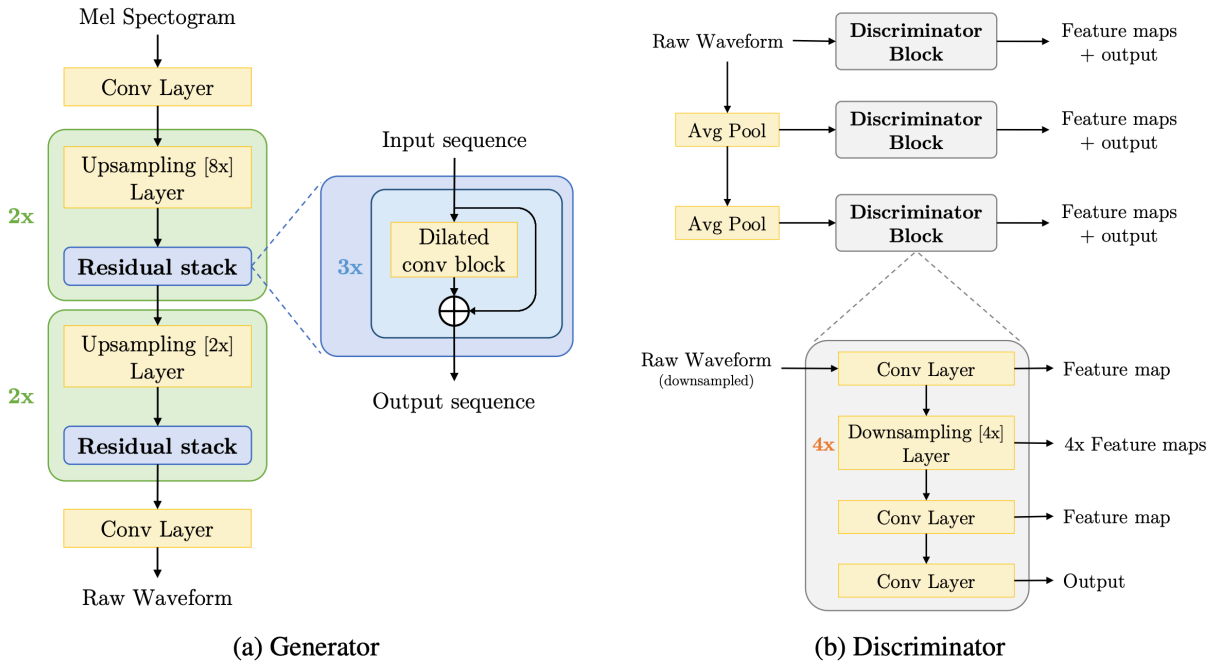

Рис. 3: Архитектура сети MelGAN. Картинка взята из [25]

Parallel WaveGAN [35] использует структуру WaveNet [2] без авторегрессии и схожую с MelGAN идею об оконных дискриминаторах, достигая не меньшего качества генерации, чем оригинальный WaveNet. VocGAN [19] предлагает накладывать дополнительные функции потерь на промежуточные слои сети, в которых содержиться информация об аудио с меньшей частотой дискритизации. MB-MelGAN [12] предлагает предсказывать multi-band сигнал аудио, тем самым резко сокращая скорость работы и колличество необходимых весов, превосходя при этом MelGAN в качестве генерации при наложении дополнительных STFT функций потерь. TFGAN [34] предлагает использование time-domain функций потерь напрямую во временной области, для повышения гарантий консистентности именно в аудио домене.

В качестве сильного бейзлайна, основанного на генеративно-состязательны сетях, мы также рассмотрим передовую модель HiFiGAN [20]. Модель оптимизирует структуру генератора MelGAN, дабавляя Multi-Receptive Field (MRF) блоки (рисунок 4), которые состоят из последовательности расширенных сверток с разными размерами ядер для лучшего анализа разнообразных паттернов в аудио.

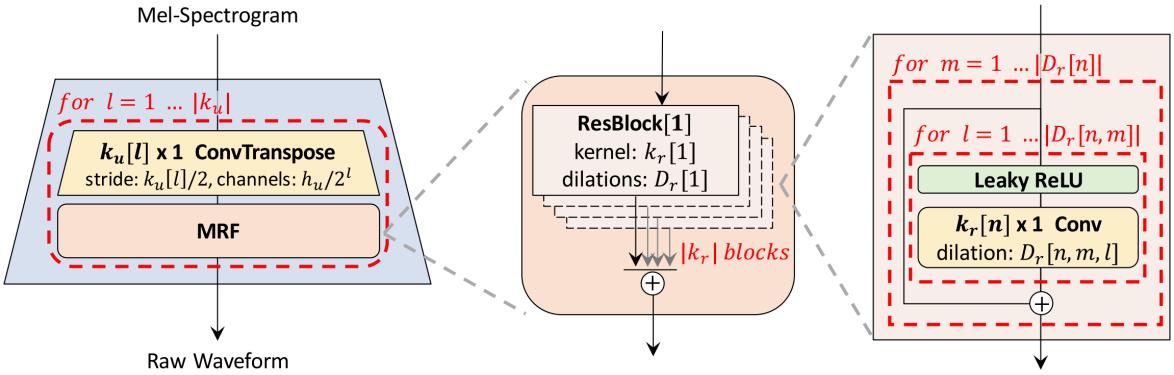

Рис. 4: Генератор HiFiGAN. Картинка взята из [20]. Модуль MRF суммирует признаки из $|k_r|$ остаточных блоков с различными размерами ядра и скоростями расширения (dilation rate).

Также к обычному Multi-Scale (MSD) дискриминатору MelGAN'а добавляется Multi-Period (MPD) тип дискриминаторов, который использует для анализа идею об волновой природе аудио. На рисунке 5 изображены различные типы дискриминаторов в HiFiGAN. MPD представляет собой смесь субдискриминаторов, каждый из которых принимает только равноотстоящие друг от друга образцы входного аудио, промежуток задается как некоторый период p. Субдискриминаторы разработаны таким образом, чтобы улавливать различные неявные структуры друг от друга, рассматривая различные части входного аудио.

HiFiGAN слегка превосходит свои авторегрессионные и Flow-подобные аналги по качеству генерации, обладая при этом крайне быстой скоростью работы. Самая маленькая версия HiFiGAN весит меньше 1M параметров, что на 2 порядка меньше модели WaveGlow с 88M параметров.

2.5. Особенности предыдущих решений

Оказывается, что качество GAN-подобных вокодеров может сильно падать на целевой задаче, поскольку базовое обучение инверсии мел-спектрограмм происходит на чистых мелах, а на этапе применения мел-спектрограмма будет приходить из какой-то целевой задачи, т.е.

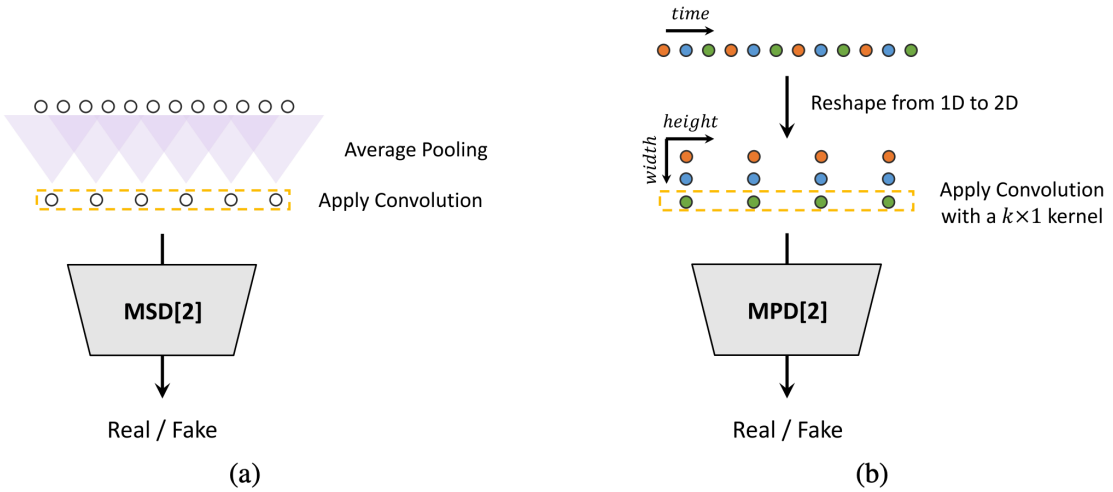

Рис. 5: Различные типы дискриминаторов в HiFiGAN. Левая (a) картинка относиться к MSD типу дискриминатора, а правая (b) к MPD дискриминатору. Картинка взята из [20].

предсказана с некой ошибкой. Для определенности в данной работе мы рассмотрим задачу замены голоса (Voice Conversion), структура которой изображена на рисунке 6, хотя предложенный способ бороться с ошибками на входах вокодера может быть обобщен для любой задачи моделирования звука в глубоком обучении, таких как Text-To-Speech или Speech Denoising. Также, на сегодняшний день, не существует универсального решения для замены голоса, которое генерировало бы чистое аудио без шумов и звуковых артефактов [21], поэтому особенно интересно и показательно будет рассмотреть устойчивость вокодеров именно на этой задаче.

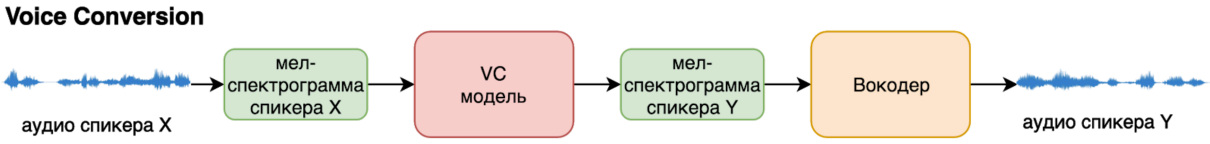

Рис. 6: Стандартная сквозная двухэтапная структура модели замены голоса. Мел-спектрограмма на выходе VC модели будет сгенерирована с ошибкой, поэтому важно чтобы вокодер обладал свойством устойчивости к этой ошибке на входе.

В таблице 1 приведены результаты автоматических метрик для GAN-подобных бейзлайнов на чистых мел-спектрограммах и пропу-

Модель	PESQ	STOI	F0-RMSE
Чистые мел-спектрограммы			
MelGAN	3.31	0.86	53.80
HiFi-GAN	3.84	0.95	36.90
Id замена голоса с помощью AutoVC[45]			
MelGAN	2.77	0.73	102.93
HiFi-GAN	3.10	0.82	87.23

Таблица 1: Автоматические метрики качества для GAN-подобных вокодеров на чистых мел-спектрограммах и с моделью замены голоса на датасете Celebrities30.

щенных через передовую модель замены голоса AutoVC [45] на датасете Celebrities30. Подробнее об используемых для оценки метриках и датасетах можно найти в соответствующем разделе 4.1. Из этой таблицы можно заметить, что задача замены голоса может вносить сильный шум в мел-спектрограмму, что приводит к серьезной деградации качества синтеза речи. На практике такое проседание означает сильные роботизированные шумы в аудио и различные звуковые артефакты.

Стандартное решение в таком случае - это до-обучить модель на выходах целевой сети, то есть на мел-спектрограммах с ошибками, но предоставив вокодеру их правильную аудио разметку. Этот способ условно можно назвать supervised, и его применимость для создания устойчивого вокодера сильно ограничена [25] по следующим причинам: 1) Ограниченность данных: если расределение ошибке в мел-спектрограмме на данном голосе сложное, а данных для этого голоса мало то модель не сможет выучить это распределение. 2) Несоответствие ошибки на мел-спектрограмме: например для задачи замены голоса правильную разметку аудио можно получить только для id замены голоса, то есть перевода голос в самого себя. Передовые модели замены голоса [45] [21] [38], как правило, дополнительно обучаются сохранять стиль и контент при id конверсии, потоэтому ошибка на них хуже соответствует целевой, т.е. при замене голоса в другого спикера.

GAN-подобные вокодеры вокодеры показывают хорошие результаты [20] качества моделирования звука и устойчивости к новым спике-

рам, имея при этом крайне быструю скорость работы и малый размер модели. Однако проблема устойчивости этого типа вокодеров к ошибке мел-спектрограммы на входе может стать серьезным препятствием для использования в реальных задачах синтеза речи, поэтому очень важно уметь правильно адаптировать вокодер к подобной ошибке. Предложенный в работе unsupervised способ дообучения будет описан в разделе 3.4, а эксперименты, показывающие превосходство данного метода над обычным supervised до-обучением, приведены в разделе 4.3.

3. Методы

В данном разделе мы подробно опишем предложенную архитеркутуру вокодера CascadeGAN для инверсии мел-спектрограмм основанной на MelGAG [25]. В параграфе 3.1 описанна структура генератора G, непосредственного осуществляющая преобразование мел-спекрограммы в аудио волну, т.е. картинки в вектор большого размера. В данной работе мы используем частоту дискретизации (sample rate) равной 22050 Гц, это означает, что в финальной аудио волне должно быть записано 22050 измерений каждую секунду аудио. Такая частота обеспечивает баланс между качеством аудио и требований к скорости обучения и быстродействию модели [18]. В разделе 3.2 мы опишем модель дискриминатора D, обучающегося отличать сгенерированное аудио от реального, и позволяющего таким образом неявно оптимизировать качество генератора через adversarial функцию потерь. В параграфе 3.3 будет описаная общая структура обучения сети и полная функция потерь, включая часть (feature matching loss) которая выступает в качестве регуляризации обучения. Затем в параграфе 3.4 мы опишем новый способ до-обучения вокодера на выходах целевой задачи (например Text-To-Speech или Voice Conversion) в unsupervised манере, не требующий правильной разметки, т.е. чистой аудио волны для зашумленных мел-спектрограмм.

3.1. Генератор

3.1.1. Общая архитектура

Структура генератора следует MelGAN [25] и состоит из последовательности полностью конволюционных слоев. Каждый слой расстягивает скрытое представление аудио по временной оси. Поскольку мел-спектрограмма это сжатое в 256 раз представление аудио (параметр hop length), то именно в 256 раз генератор и растягивает входную акустичекую характеристику. Мы обнаружили, что оптимальное распределение коэффициентов растягивания в таком случае выглядит как:

x8, x8, x2, x2, т.е. генератор состоит из 4 слоев.

В отличие от MelGAN, теперь генератор обрабатывает скрытые представления не последовательно слой за слоем, а анализирует выходы всех предыдущих слоев на каждом шаге, чтобы произвести очередную дискретизацию. Мы назвали такой слой Cascade Block и подробнее он будет описан в следующем разделе.

Следуя VocGAN [19], мы также накладыаем дополнительные функции потерь не только на финальный выход сети, но и на выходы 2 и 3 слоев, которые соответствуют представлению аудио с в 4 и 2 раза меньшей частоте дискретизации соответственно. Таким образом каждый выход нашей сети на этапе обучения содержит кортеж из 3 аудио волн, соответствующих 5512 Гц, 11025 Гц, 22050 Гц частоте дискретизации. На этапе тестирования, используется только финальный выход. Такой трюк помогает поддерживать более широкую структуру аудио и учитывать долгосрочные волновые зависимости при обучении.

3.1.2. Cascade Block

В оригинальном MelGAN'е локальное растягивание аудио представления не гарантирует согласованности на больших масштабах. Идея Cascade Block улучшить такую согласованность, проанализировав аудио на всех масштабах на очередном слое. Мы также используем Multi-Receptive Field (MRF) блоки из модели HiFi-GAN [20] для анализа разнообразных паттернов в аудио представленнии. Обозначим $\mathrm{MRF}^{b_1,b_2,\ldots}_{a_1,a_2,\ldots}$ как блок содержащий $a_1, a_2, \ldots$ как размеры ядер и $b_1, b_2, \ldots$ как скорости расширения (dilation rates). Подробнее об этом блоке можно узнать из рисунка 4 или в статье [20].

На рисунке 7 изображена архитектура Cascade Block, всего же в сети 4 таких блока. Cascade Block принимает на вход все выходы предыдущих слоев, соответствующих представлению аудио более низкой размерности, растягиет каждый из них до единого масштаба, затем пропускает через $\mathrm{MRF}^{3,5,7}_{3,5}$ блок и наконец суммирует полученные представления. Для лучшего анализа баланса между масштабами при суммировании, получившееся представление проходит через допол-

нительных $\mathrm{MRF}_{7,11}^{3,5,7,11}$ блок, с большим колличеством разнообразных расширенных сверток и размеров ядер. Контролируя гиперпараметры MRF блоков в разных частях модели, можно добиваться баланса между скоростью и лучшим анализом аудио характеристик. Тут стоит отметить, что Cascade Block работал бы сильно дольше оригинального MelGAN, если не была произведена замена Deconv блоков, которые составляли большую часть весов GAN-подобных вокодеров и дольше всего обрабатывали внутренние представления. Подробнее об этом написанно в следующем разделе.

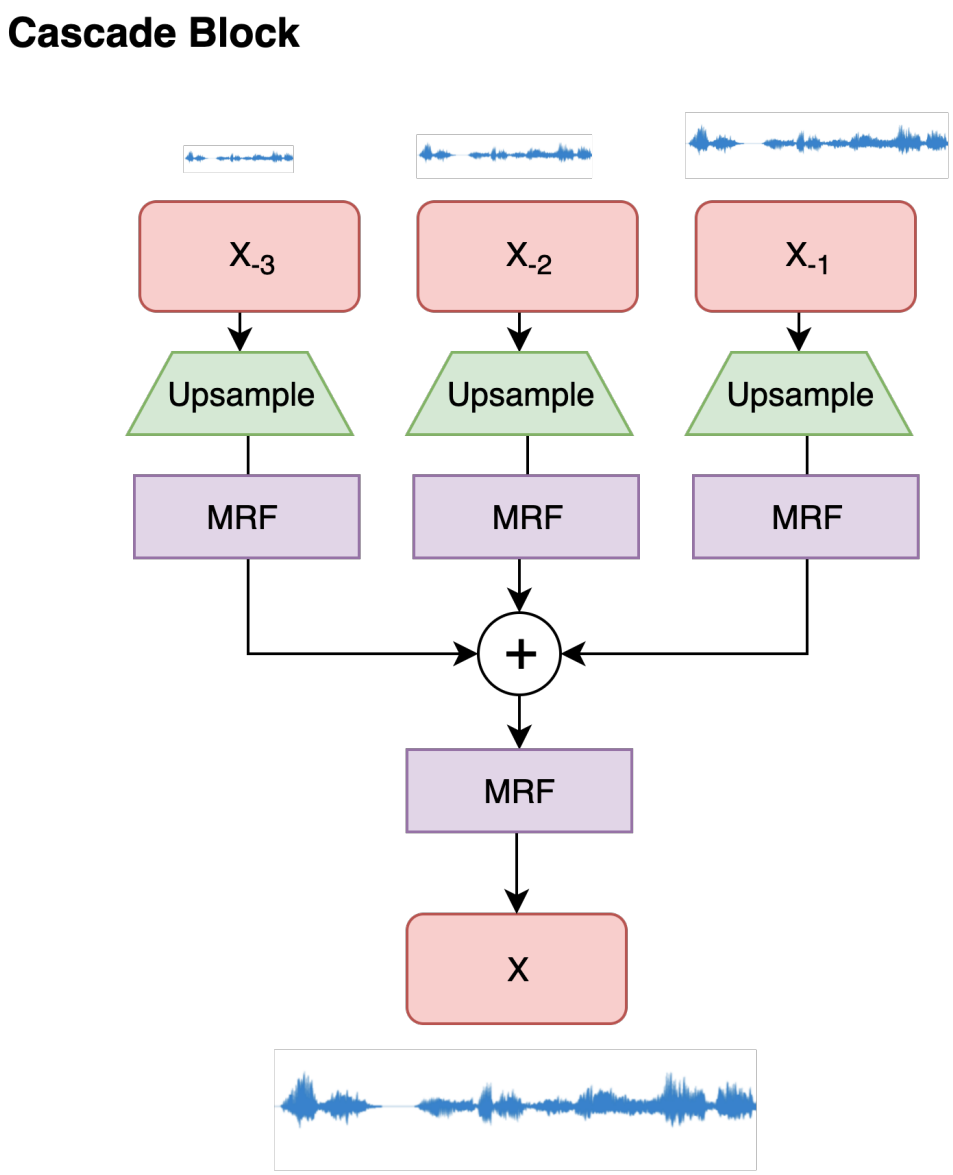

Рис. 7: Архитектура Cascade Block. Multi-Receptive Field (MRF) это модифицированные блоки из модели HiFi-GAN [20]. X - текущий выход блока, X_{-n} - предыдущие выходы, соответствующие низко размерному представлению аудио.

3.1.3. Замена Deconv блоков

В работе [31] показано, что стандартный подход к получению изображений с помощью деконволюции, несмотря на все его успехи, имеет некоторые концептуально простые проблемы, которые приводят к так называемым шахматным артефактам в полученных изображениях, появляющиеся из-за сильного перекрытия внутренних рецептивных полей (см. рисунок 8). Использование естественной альтернативы

без этих проблем приводит к исчезновению артефактов. В данной работе мы решили применить эту идею и к задаче синтеза звука, заменив Deconv слой на UpsampleNearest, за которым следует простой сверточный слой и который для краткости будем называть просто Upsample. Такая замена преследует сразу две цели: 1) Устранение артефактов, которые возникают при использовании Deconv слоев. 2) Облегчение общей структуры генератора и оптимизация скорости вычислений.

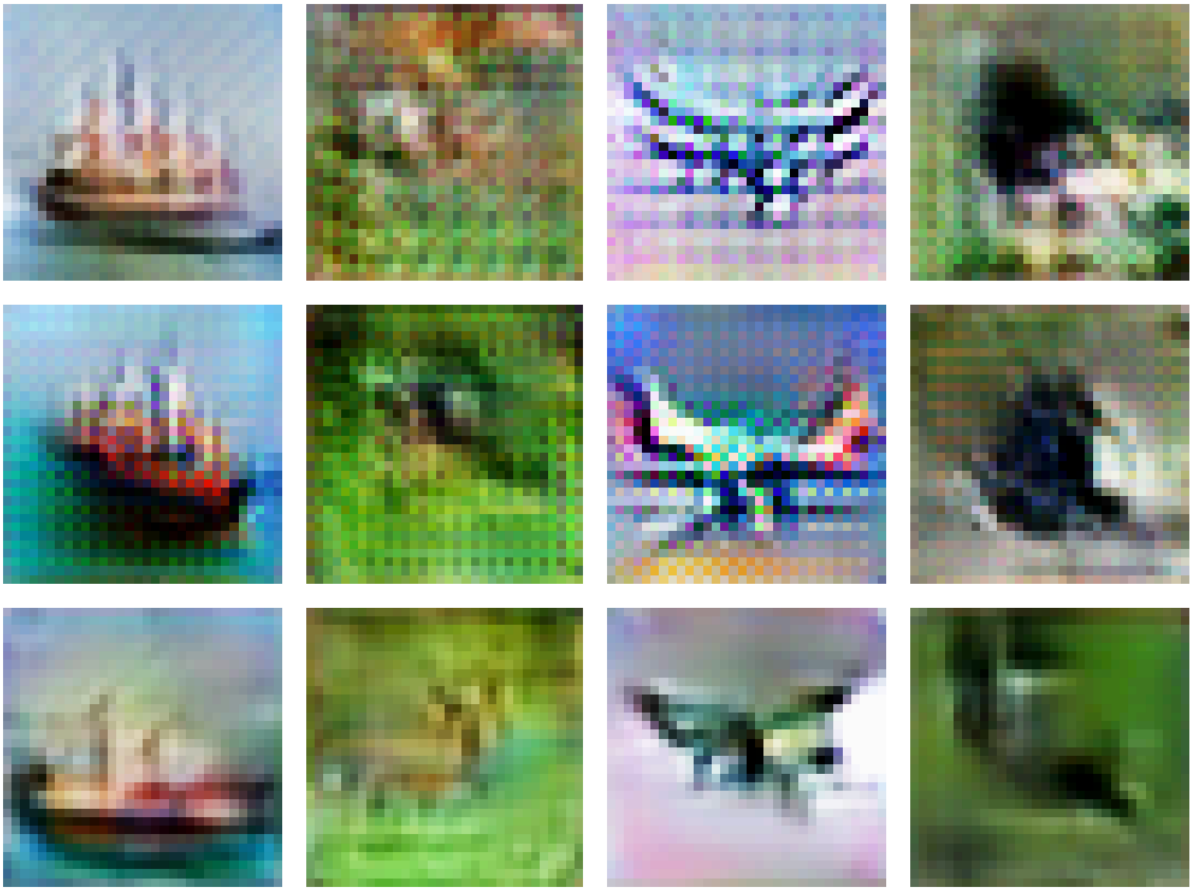

Рис. 8: Пример шахматных артефактов на задаче генерации изображений. Картинка взята из [31]. Верхняя строка - деконволюция в двух последних слоях, артефакты частоты 2 и 4. Средняя строка - деконволюция только в последнем слое, артефакты частоты 2. Нижняя строка - все слои Upsample, нет артефактов.

3.1.4. Функция потерь

Функция потерь для генератора, помимо adversarial части, которая будет описанна в части 3.3, состоит из 3 типов: 1) STFT loss 2) Time

loss 3) Mel loss. Эти функции потерь используются как стабилизирующие процесс обучения генеративно-состязательных сетей и улучшают их эффективность. Стоит отметить, что как показано в [25], использование одних только стабилизирующих функций потерь недостаточно и такое обучение всегда будет приходить к роботизированному звуку или иным аудио артефактам. Вкратце опишем каждый из этих типов функций потерь.

Оценка различия между особенностями истинной и сгенерированной речи представляет особенную трудность. Без внедрения подобной оценки в функцию потерь, процеес сходимости окажется чрезвычайно медленным и не стабильным. Для решения подобной проблемы впервые были предложены Multi-resolution STFT функции потерь [35], которые для краткости мы будем называть STFT loss. Пусть x - чисто аудио, а $\widetilde{x}$ - сгенерированное, соответствующее мел-спектрограмме x. Тогда для каких-то выбранных единых параметров STFT:

$$L_{sc}(x, \widetilde{x}) = \frac{\| |STFT(x)| - |STFT(\widetilde{x})| \|_F}{\|STFT(x)\|_F} \tag{1}$$

$$L_{mag}(x, \widetilde{x}) = \frac{1}{N} \| \log |STFT(x)| - \log |STFT(\widetilde{x})| \|_1 \tag{2}$$

где $\|.\|_F$ и $\|.\|_1$ - это нормы Фробениуса и ℓ_1 соответственно. Для увеличения разнообразия паттернов и масштабов структур, которые вовлечены в данную функцию потерь, мы проссумируем данные функции потерь как для разных наборов параметров STFT (всего их M и параметры взяты из [35]), так и для всех выходов нашей сети, соответствующих аудио с разной частотой дискретизации:

$$L_{\text{STFT}} = \mathop{\mathbb{E}}_{x,\widetilde{x}} \left(\frac{1}{3M} \sum_{m=1}^{M} \sum_{p=1}^{3} L_{sc}^{m}(x_p, \widetilde{x}_p) + L_{mag}^{m}(x_p, \widetilde{x}_p) \right) \tag{3}$$

где m - это набор параметров для STFT, p - номер выхода сетки.

Следуя модели TFGAN [34], мы также используем time-domain функции потерь для кадого из выходов сети, который для краткости будем называть Time loss. Эти функции потерь действуют анало-

гично STFT и проходятся по аудио с некотором размером окна (frame length) и размером шага (hop length), наборы параметров для которых мы взяли из [34] (всего их S). Однако вместо перевода в частотную область, считается некоторая статистика над набором измерений аудио в окне:

$$L_e(x, \widetilde{x}) = \| \mathbb{E}(x^2) - \mathbb{E}(\widetilde{x}^2)\|_1 \tag{4}$$

$$L_t(x, \widetilde{x}) = \| \mathbb{E}(x) - \mathbb{E}(\widetilde{x})\|_1 \tag{5}$$

$$L_p(x, \widetilde{x}) = \| \mathop{\mathbb{E}}_t(x_t - x_{t-1}) - \mathop{\mathbb{E}}_t(\widetilde{x}_t - \widetilde{x}_{t-1})\|_1 \tag{6}$$

$$L_{\text{Time}} = \mathop{\mathbb{E}}_{x,\widetilde{x}} \left(\frac{1}{3S} \sum_{s=1}^{S} \sum_{p=1}^{3} L_e^s(x_p, \widetilde{x}_p) + L_t^s(x_p, \widetilde{x}_p) + L_p^s(x_p, \widetilde{x}_p) \right) \tag{7}$$

где s - это набор параметров для размеров окна и шага, p - номер выхода сетки. Такая функция потерь дает повышение гарантий консистентности непосредственно в аудио домене.

Также следуя HiFiGAN [20], мы накладываем на финальный выход нашей сети дополнительную функцию потерть в частотной области в мел-шкале или Mel loss. Пусть $\phi(.)$ это функция мел-преобразования. Тогда функция потерь для консистентности мел-инверсии выглядит как:

$$L_{\text{Mel}}(x, \widetilde{x}) = \mathop{\mathbb{E}}_{x,\widetilde{x}} \|\phi(x) - \phi(\widetilde{x})\|_1 \tag{8}$$

Такая функция потерь уделяет больше внимания улучшению качества восприятия из-за особенностей слуховой системы человека и помогает генератору синтезировать реалистичную звуковую волну, соответствующую входным условиям, а также стабилизирует процесс обучения с ранних этапов.

3.2. Дискриминатор

3.2.1. Типы дискриминаторов

Цель дискриминаторов - уметь эффектино различать реальное аудио от сгенерированного данным состоянием генератора в процессе обучения. При этом чем более разнообразны масштабы рассмотрения [25] и извлекаемые паттерны [20], тем лучше затем adversarial часть функции потерь генератора будет приближать качество генерации к реальному и устранять различные шумы и артефакты. Для разнообразия извлекаемой из аудио информации в работе мы используем сразу 3 разных вида дискриминаторов: 1) Multi-Scale дискриминаторы из MelGAN [25] - анализируют аудио на разных масштабах, предварительно понижая дискретизацию, а также промежуточные выходы сети, соответсвующие меньшей частоте дискретезации. 2) Multi-Period дискриминаторы из HiFi-GAN [20] - множество субдискриминаторов, анализирующие аудио через какой-то заданный период. 3) Mel-Based дискриминатор, накладываемый только на финальный выход сети и анализирующий его мел-спектрограмму. Каждый из дискриминаторов обладает window-based структурой, т.е. на выходе выдает вектор чисел размером пропорциональный временной оси, где значения локально соответствуют качеству аудио в этом регионе.

Такой пестрый набор дискриминаторов повышает дискриминативную способность модели и позволяет подмечать разнообразные структурные различия качества реального аудио и снегерированного моделью, тем самым повышая качество модели через adversarial функцию потерь дискриминатора без необходимости вычисления схожести аудио напрямую.

3.2.2. Обуславливание на мел-спектрограмме

Для лучшего различия real/fake аудио, каждый дискриминатор обуславливается на исходную мел-спектрограмму. На рисунке 9 изображена схема любого из таких субдискриминаторов. Для Multi-Period и Mel-Based дискриминаторов на этом рисунке входное аудио предва-

рительно предобрабатывается в соответствии с анализируемой в дискриминаторе структурой. Сначала исходная мел-спектрограмма расширяется по временной оси, в соответствии с внутренним представлением дискриминатора, затем она конкатенируется с этим представлением и анализируется дальше. Такая согласованность по временной оси помогается лучше локализовывать и различать структурные особенности представлений для разных входов дискриминаторов.

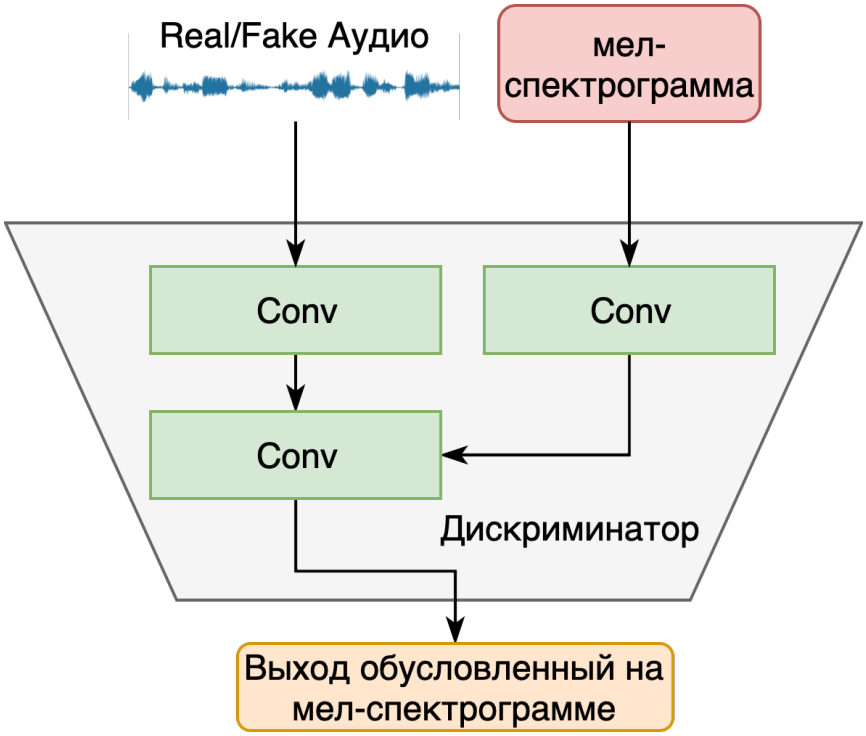

Рис. 9: Схема обуславливания на мел-спектрограмме. Исходная мел-спектрограмма расширяется по временной оси, в соответствии с внутренним представлением дискриминатора, затем она конкатенируется с этим представлением и анализируется дальше.

3.2.3. U/V коррекция

Используемые window-based дискриминаторы обновляют функцию потерь даже в регионах, где спикер мог сделать достаточно большую паузу. При этом информации в этом регионе недостаточно, чтобы с уверенностью говорить о качестве аудио, а adversarial часть функции ошибки генератора будет вносить ненужный шум. На рисунке 10 изображена схема решения данной проблемы, предложенная в статье.

По входному аудио вычисляется U/V бинарная маска, с регионами где была произнесена речь. Далее эта маска интерполируется по временной оси выхода дискриминатора и используется, чтобы обновить функцию потерь только в регионах голоса.

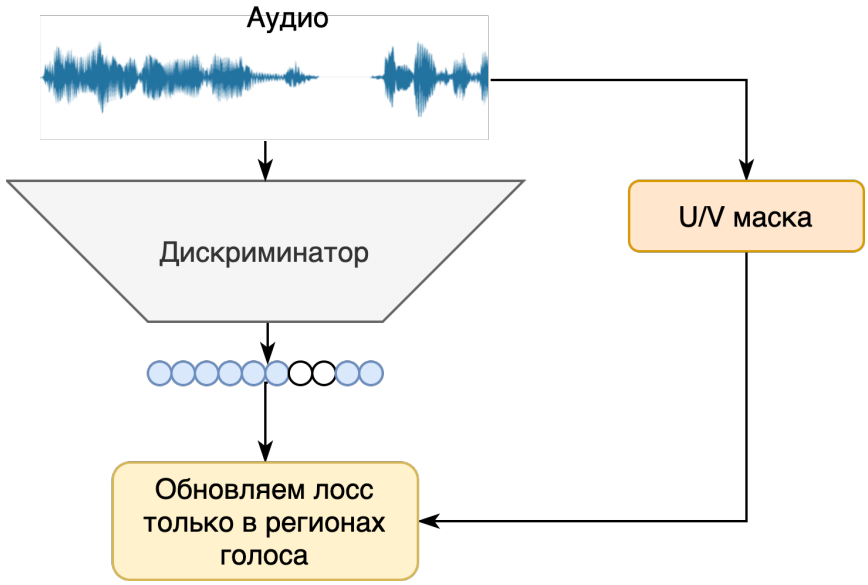

Рис. 10: Принцип U/V коррекции. По входному аудио вычисляется U/V бинарная маска, которая затем используется чтобы обновить функцию потерь только в регионах голоса.

3.2.4. Функции потерь

Для генератора и дискриминатора цели обучения соответствуют LSGAN [42], которые заменяют бинарные условия перекрестной энтропии оригинальных целей GAN [16] функциями потерь наименьших квадратов для не убывающих градиентных потоков. Дискриминатор обучается классифицировать аудио образцы, полученные в качестве чистых аудио, на 1, а аудио, синтезированные из генератора, на 0. Генератор обучается обманывать дискриминатор, обновляя качество аудио для классификации до значения равного 1. Функции потерь для генератора G и дискриминатора D определяются как:

$$L_{\text{Adv}}^{D} = \mathop{\mathbb{E}}_{x,\widetilde{x}} \left(\frac{1}{Q} \sum_{q=1}^{Q} (D_q(x) - 1)^2 + (D_q(\widetilde{x}))^2 \right) \tag{9}$$

$$L_{\text{Adv}}^{G} = \mathop{\mathbb{E}}_{x,\widetilde{x}} \frac{1}{Q} \sum_{q=1}^{Q} (D_q(\widetilde{x}) - 1)^2 \tag{10}$$

где q - тип дискриминатора с соответвующем ему типом предобработки входного аудио (понижение частоты дискретизации, мел-преобразование или переодизация), Q - число разных дискриминаторов.

3.3. Общая структура обучения модели

Feature matching функция потерь - это метрика сходства, измеряемая разницей в признаках дискриминатора между истиным и сгенерированным аудио образцом [3] [25]. Поскольку он был успешно применен для синтеза речи [25], мы используем его в качестве дополнительной функции потери для стабилизации обучения генератора. Каждый промежуточный признак дискриминатора извлекается, и в каждом пространстве признаков вычисляется расстояние ℓ_1 между истинным и сгенерированным аудио. Feature matching loss определяется как:

$$L_{\text{Fm}} = \mathop{\mathbb{E}}_{x,\widetilde{x}} \left(\frac{1}{Q} \sum_{q=1}^{Q} \sum_{i=1}^{T} \| D_q^i(x) - D_q^i(\widetilde{x}) \|_1 \right) \tag{11}$$

где T - это число слоев в дискриминаторе D_q, а i - номер слоя.

Следуя [12], для стабилизации процесс обучения модели происходит в 2 этапа: 1) Обучение только G без adversarial функций потерь. 2) Обучение и G и D с полной функцией потери, включая adversarial часть. При этом 1-я часть обучения ответственна за правильное отображение из мел-спектрограммы в аудио на сравнительно широких масштабах, а 2-я - это тонкая настройка мелкодисперсного рисунка в аудио, ответственного за наличие или отсутствие разнообразных шумов или аудио артефактов. Полная функция потерь для генератора и дискриминатора выглядят как:

$$L^{D} = \lambda_{\text{Adv}} L_{\text{Adv}}^{D}(x, \widetilde{x}) \tag{12}$$

$$L^G = \lambda_{\text{Adv}} L^G_{\text{Adv}}(\widetilde{x}) + \lambda_{\text{Fm}} L_{\text{Fm}}(x, \widetilde{x}) + L_{\text{Mel}}(x, \widetilde{x}) + L_{\text{Time}}(x, \widetilde{x}) + L_{\text{STFT}}(x, \widetilde{x}) \quad (13)$$

при этом, λ_{Adv} и λ_{Fm} первую часть обучения равны нулю.

3.4. Unsupervised до-обучение модели

Оказывается, что с помощью структуры GAN'ов можно не только устранять шумы и артефакты в аудио, но изящно решить проблему зашумленных мелов. Идея в том, что до-обучить модель на спектрограммах без разметки, тем самым внедрив в дискриминатор информацию об ошибки с уровня модели на уровень данных. Преимущество в том, что таких спектрограмм можно получить много больше, а ошибка на них лучше соответствует целевой. Например, для Voice Conversion голоса такие мел-спектрограммы можно получить выполняя конверсию голоса типа любой→любой, а для задачи Text-To-Speech подавая на вход произвольный связный текст и произвольного спикера. Будем также до-обучать и сам дискриминатор, поскольку в конце обучения на чистых мелах его дискриминативная способность по различию хорошего качества аудио уже велика, достаточно будет научить его находить ошибку не только в модели, но в шумных данных.

Пусть для некоторой целевой задачи мы подготовили два датасета зашумленых мелов: 1) M_{sup} содержащий мел-спектрограммы с ошибкой и соответствующие им правильные аудио и 2) M_{upsup} содержащий только мел-спектрограммы с ошибкой, т.е. для которых нет правильного аудио. При этом каждую итерацию до-обучения, мы имеем $\widetilde{x}_{\text{sup}}$, x_{sup} и $\widetilde{x}_{\text{unsup}}$ - выход генератора на M_{sup}, правильное аудио на M_{sup} и выход генератора на M_{upsup} соответственно. Тогда формулы 14 и 15 поменяются следующим образом:

$$L^D = \lambda_{\text{Adv}} L^D_{\text{Adv}}(x_{\text{sup}}, \widetilde{x}_{\text{unsup}}) \quad (14)$$

$$L^G = \lambda_{\text{Adv}} L^G_{\text{Adv}}(\widetilde{x}_{\text{unsup}}) + \lambda_{\text{Fm}} L_{\text{Fm}}(x_{\text{sup}}, \widetilde{x}_{\text{sup}}) + L_{\text{Mel}}(x_{\text{sup}}, \widetilde{x}_{\text{sup}}) + L_{\text{Time}}(x_{\text{sup}}, \widetilde{x}_{\text{sup}}) + L_{\text{STFT}}(x_{\text{sup}}, \widetilde{x}_{\text{sup}}) \quad (15)$$

4. Эксперименты

4.1. Описание экспериментов

4.1.1. Используемые метрки

Метрики, используемые в данной работе, можно поделить на обьективные и субьективные, как показанно на рисунке 11.

MOS - субьективная метрика, которая использует оценку ассесорами на Краудсорсинг платформах, таких как Amazon Mechanical Turk или Yandex Toloka (использована в работе). Все сгенерированные образцы не были показаны во время обучения. MOS были подсчитаны на популяции из 200 человек: каждого из них попросили вслепую оценить подмножество из 15 аудио образцов, взятых случайным образом из этого пула образцов, оценивая образцы от 1 до 5 баллов. Мы потребовали, чтобы участники тестирования использовали наушники и владели английским языком. После сбора всех оценок, MOS-балл μ_i модели i оценивается путем усреднения оценок m_i для аудио образцов, для различных типов моделей. Кроме того, мы вычисляем 95% доверительный интервал, где σ^i - стандартное отклонение собранных оценок:

$$\hat{\mu}_i = \frac{1}{N_i} \sum_{k=1}^{N_i} m_{i,k} \tag{16}$$

$$\mathrm{CI}_i = \left[\hat{\mu}_i - 1.96 \frac{\hat{\sigma}_i}{\sqrt{N_i}}, \hat{\mu}_i + 1.96 \frac{\hat{\sigma}_i}{\sqrt{N_i}} \right] \tag{17}$$

Другой тип метрик - это обьективные или автоматические метрики. Они включают: Perceptual Evaluation of Speech Quality (PESQ), Short-Time Objective Intelligibility (STOI) и F_0 Root Mean Square Error F0-RMSE которые измеряют уровень шумов, понимаемость речи и предсказание питча (фундаментальной частоты) голоса соответственно. PESQ [1] варьирутеся от -0.5 до 4.5, чем он выше тем лучше. STOI [5] варьирутеся от 0 до 1, чем он выше тем лучше. И наконец, F0-RMSE [4] оценивает предсказание функдаментальной частоты, чем

ниже эта метрика тем лучше.

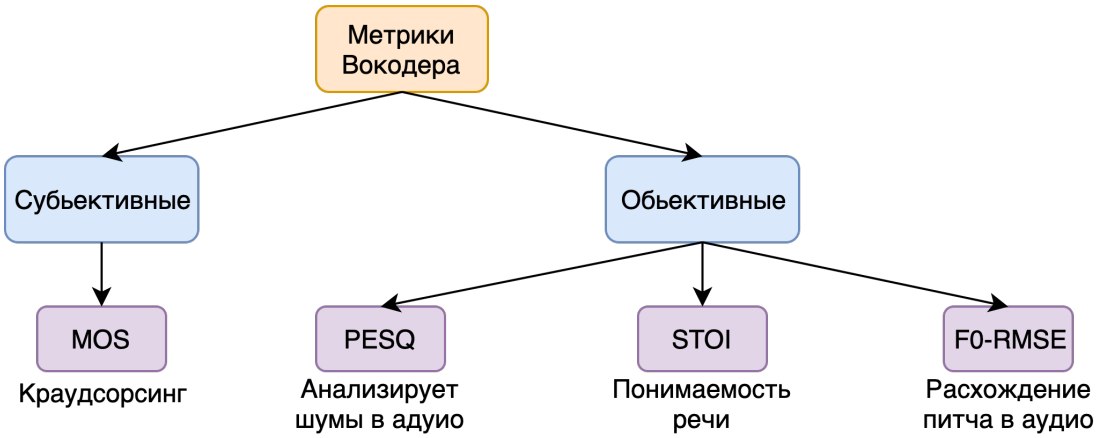

Рис. 11: Типы и разновидности используемых в работе метрик.

4.1.2. Используемые датасеты

Два датасета была использовано для оценки качества генерации голоса в данной работе: 1) LibriTTS и 2) Celebrities30. Все используемые аудио образцы были передискретизированны в 22050 Гц.

Для оценки качества на разных типах спикеров мы использовали Multi-Speaker датасет LibriTTS [15], содержащий 585 часов речевых данных для 2456 спикеров. Для простоты мы использовали подвыборку train-clean-100 для обучения, поскольку она содержит наиболее чистые по акустике образцы. Для тестирования была использована подвыборка test-clean.

Для оценки моделей на целевой задаче замены голоса в приближенных к реальным условиях, мы собрали дополнительный датасет из 30 спикеров Celebrities30, использовав Speaker Encoder из [26] чтобы вырезать из платформы YouTube по 1 часу для каждого спикера. 10 минут для каждого спикера были использованы для тестирования, остальное - для обучения. Для лучшей адаптируемости к малому числу спикеров для всех моделей тестируемых на Celebrities30, мы также добавили в обучающую выборку и LibriTTS train-clean-100 как дополнительный набор данных.

4.1.3. Особенности обучения моделей

CascadeGAN обучается в 2 этапа, по 500 тысяч тренированых шагов каждый. Мы использовали разер батча равный 16 и 4 GPU. Все остальные модели были имплементированы в соответсвии с используемыми статьями или взяты из открытых источников.

Для дообучения на выходах целевой задачи замены голоса мы использовали AutoVC [45], обученный на LibriTTS и Celebrities30. Затем CascadeGAN до-обучается еще 200 тысяч тренировочных шагов на соответствующих выходах. При этом, чтобы корректно измерить автоматические метрики, мы используем id-конверсию голоса на этапе тестирования и честную замену голоса в другого спикера для оценки MOS.

4.2. Результаты экпериментов

В таблице 2 представлены автоматические метрики качества и скорости работы на чистых входах для бейзлайнов и предложенной в работе модели CascadeGAN на датасете LibriTTS. Также представлены результаты для CascadeGAN без совокупности изменений генератора G и дискриминатора D относительно базового MelGAN [25]. CascadeGAN повысил метрику PESQ на 0.06, метрику STOI на 0.01, а метрику F0-RMSE понизил на 4.51 относительно лучшей модели бейзлайнов HiFiGAN. Также, однако, можно заметить, что CascadeGAN имеет в 2 раза больший вес и в 2 раза меньшее быстродействие работы, чем HiFiGAN. Происходит это из-за особенностей каскадной структуры модели, когда выходы предыдущих слоев много раз повторно переизпользуются для дальнейшего анализа. Стоит отметить, что, и хотя скорость работы предложенной модели ниже, она все равно почти в 5 раз превышает real-time генерацию звука, и подобное снижение можно считать допустимым компромисом между скоростью и качеством работы. В целом, CascadeGAN превосходит бейзлайны по всем характеристикам качества, оставаясь легковесным и быстрым на этапе применения.

Модель	PESQ	STOI	F0-RMSE	Размер (M)	Скорость на CPU (kHz)
WaveNet	3.78	0.93	32.12	24.7	0.06
WaveGlow	3.73	0.90	33.85	87.9	1.58
MelGAN	3.40	0.89	41.80	4.26	51.9
HiFi-GAN	3.87	0.96	33.36	**0.92**	**214.97**
CascadeGAN	**3.93**	**0.98**	**28.85**	1.94	103.6
w/o G-изменения	3.64	0.84	36.39	4.26	51.9
w/o D-изменения	3.79	0.91	39.03	1.94	103.6

Таблица 2: Метрики качества, размер модели и быстродействие на LibriTTS датасете для предыдущих моделей и для предложенной в работе CascadeGAN.

4.3. Результаты экспериментов до-обучения на мел-спектрограммах с ошибкой

В таблице 3 представлены результаты экпериментов для работы вокодеров на выходах целевой задачи замены голоса для модели AutoVC. Все модели дополнительно supervised до-обучены на мел-спектрограммах с соответствующими им чистыми аудио. CascadeGAN также дополнительно был unsupervised до-обучен по схеме описанной в разделе 3.4. Из таблицы мы видим, что новый способ до-обучения сильно повысил как субьективные, так и все обьективные метрики качества на задаче замены голоса. CascadeGAN статистически значимо повысил метрику MOS на 0.49 относительно лучшей предыдущей модели WaveGlow, обладающей природной устойчивостью к зашумленным мелам [11]. Также, автоматические метрики PESQ, STOI и F0-RMSE улучшены на 0.47, 0.08 и 21.61 соответственно. В целом, до-обучение на зашумленных мелах без разметки сильно повысило метрики качества и тем самым устройчивость GAN-подобных вокодеров к ошибке на входе.

Модель (все supervised ft)	MOS	PESQ	STOI	F0-RMSE
WaveNet	2.89 (±0.13)	2.72	0.68	113.11
WaveGlow	3.41 (±0.10)	3.29	0.85	63.94
MelGAN	2.77 (±0.12)	2.81	0.73	102.93
HiFi-GAN	3.10 (±0.09)	2.94	0.82	87.23
CascadeGAN	3.13 (±0.09)	3.05	0.81	78.91
CascadeGAN + unsupervised ft	**3.90 (±0.09)**	**3.76**	**0.94**	**42.25**

Таблица 3: Метрики качества для задачи Voice Conversion с моделью AutoVC на датасете Celebrities30. (un)supervised ft - тип до-обучения.

Выводы

В работе предложена новая каскадная архитектура для GAN-подобного генератора аудио, которая лучше согласовывает структуру аудио с его более низкоразмерным представлением: мел-спектрограммой или выходами предыдущих слоев генератора с меньшим масштабом. Новая архитектура дискриминатора состоит из 3 частей с разным типом входа, что призвано разнообразить информацию для анализа аудио и улучшить дискриминативную способность модели. Модель превосходит бейзланы по метрикам качества, повышая PESQ на 0.06 на датасете LibriTTS.

Также в работе разработан и исследован новый unsupervised способ адаптации моделей к ошибке на входе, не требующий правильной аудио разметки и поэтому являющийся общим для широкого класса задач. Проведенные эксперименты показывают, что до-обучение для модели CascadeGAN на целевой задаче замены голосы повысило метрику MOS на 0.49, убрав значительное число аудио артефактов и шумов.

По итогам данной работы, было показано что GAN вокодеры на сегодняшний день остаются лучшими для задачи генерации звука, показывая лучшее качество и скорость работы в несколько раз превышающий реальное время на CPU. В работе показано, что не существует серьезных препятствий при использовании GAN вокодеров на практике при сквозном двухэтапном синтезе аудио. И хотя предложенный способ до-обучения моделей является общим и требует только зашум-

ленных мел-спектрограмм, дальнейшие работы, обобщающие эту технику для более широкого класса моделей, в этом направлении еще предстоят.

Список литературы

[1] A. W. Rix J. G. Beerends M. P. Hollier, Hekstra. A. P. Perceptual evaluation of speech quality (pesq)-a new method for speech quality assessment of telephone networks and codecs. // IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP). — 2001.

[2] Aaron van den Oord Sander Dieleman Heiga Zen Karen Simonyan Oriol Vinyals Alex Graves Nal Kalchbrenner Andrew Senior, Kavukcuoglu. Koray. WaveNet: A Generative Model for Raw Audio. // arXiv preprint arXiv:1609.03499. — 2016.

[3] Anders Boesen Lindbo Larsen Søren Kaae Sønderby Hugo Larochelle, Winther. Ole. Autoencoding beyond pixels using a learned similarity metric. // International conference on machine learning. — 2016.

[4] Bäckström Tom. Fundamental frequency. // https://wiki.aalto.fi/pages/viewpage.action?pageId=149890776. — 2020.

[5] Cees H. Taal Richard C. Hendriks Jesper Jensen. A short-time objective intelligibility measure for time-frequency weighted noisy speech. // 2010 IEEE International Conference on Acoustics, Speech and Signal Processing. — 2010.

[6] Chan C. Ginosar S. Zhou T., Efros A. A. Everybody dance now. // arXiv preprint arXiv:1808.07371. — 2018.

[7] Davis S., Mermelstein P. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. // IEEE Transactions on Acoustics, Speech and Signal Processing. — 1980. — P. vol. 28, no. 4, pp. 357 – 366.

[8] Engel J. Agrawal K. K. Chen S. Gulrajani I. Donahue C., Roberts A. Gansynth. Adversarial neural audio synthesis. // arXiv preprint arXiv:1902.08710. — 2019.

[9] Engel J. Resnick C. Roberts A. Dieleman S. Norouzi M. Eck D., Simonyan K. Neural audio synthesis of musical notes with wavenet autoencoders. // Proceedings of the 34th International Conference on Machine Learning-Volume 70. — 2017. — P. pp. 1068–1077.

[10] Engel J. Resnick C. Roberts A. Dieleman S. Norouzi M. Eck D., Simonyan K. Clarinet: Parallel wave generation in end-to-end text-to-speech. // arXiv preprint arXiv:1807.07281. — 2018.

[11] Engel J. Resnick C. Roberts A. Dieleman S. Norouzi M. Eck D., Simonyan K. Waveglow: A flow-based generative network for speech synthesis. // ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing. — 2019. — P. pp. 3617–3621.

[12] Geng Yang Shan Yang Kai Liu Peng Fang Wei Chen Lei Xie. Multi-band MelGAN: Faster Waveform Generation for High-Quality Text-to-Speech. — 2020.

[13] Griffin D., Lim J. Signal estimation from modified short-time fourier transform. // IEEE Transactions on Acoustics, Speech, and Signal Processing. — 1984. — P. 32(2):236–243.

[14] Gulrajani I. Ahmed F. Arjovsky M. Dumoulin V., Courville A. C. Improved training of wasserstein gans. // Advances in Neural Information Processing Systems. — 2017. — P. pp. 5767–5777.

[15] Heiga Zen Viet Dang Rob Clark Yu Zhang Ron J. Weiss Ye Jia Zhifeng Chen Yonghui Wu. LibriTTS: A Corpus Derived from LibriSpeech for Text-to-Speech. — 2019.

[16] Ian J. Goodfellow Jean Pouget-Abadie Mehdi Mirza Bing Xu David Warde-Farley Sherjil Ozair Aaron Courville Yoshua Bengio. Generative Adversarial Networks. // arXiv:1406.2661. — 2014.

[17] Isola P. Zhu J.-Y.-Zhou T., Efros A. A. Image-to-image translation with conditional adversarial networks. // Proceedings of the IEEE conference on computer vision and pattern recognition. — 2017.

[18] J. Shen R. Pang R. Weiss M. Schuster N. Jaitly Z. Yang-Z. Chen Y. Zhang Y. Wang R. Skerrv-Ryan R. Saurous Y. Agiomvrgiannakis, Wu Y. Natural tts synthesis by conditioning wavenet on mel spectrogram predictions. // aIEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). — 2018. — P. pp. 4779–4783.

[19] Jinhyeok Yang Junmo Lee Youngik Kim Hoonyoung Cho Injung Kime. VocGAN: A High-Fidelity Real-time Vocoder with a Hierarchically-nested Adversarial Network. — 2020.

[20] Jungil Kong Jaehyeon Kim Jaekyoung Bae. HiFi-GAN: Generative Adversarial Networks for Efficient and High Fidelity Speech Synthesis. — 2020.

[21] Kaizhi Qian Zeyu Jin Mark Hasegawa-Johnson Gautham J. Mysore. F0-consistent many-to-many non-parallel voice conversion via conditional autoencoder. — 2020.

[22] Kalchbrenner N. Elsen E. Simonyan K.-Noury S. Casagrande N. Lockhart E.-Stimberg F. Oord A. v. d. Dieleman S., Kavukcuoglu K. Efficient neural audio synthesis. // arXiv preprint arXiv:1802.08435. — 2018.

[23] Karras T. Aila T. Laine S., Lehtinen J. Progressive growing of gans for improved quality, stability and variation. // arXiv preprint arXiv:1710.10196. — 2017.

[24] Kingma D. P., Dhariwal P. Glow: Generative flow with invertible 1x1 convolutions. // Neural Information Processing Systems. — 2018. — P. pp. 10215–10224.

[25] Kundan Kumar Rithesh Kumar Thibault de Boissiere Lucas Gestin Wei Zhen Teoh-Jose Sotelo Alexandre de Brebisson Yoshua Bengio Aaron Courville. MelGAN: Generative Adversarial Networks for Conditional Waveform Synthesis. — 2019.

[26] Li Wan Quan Wang Alan Papir Ignacio Lopez Moreno. Generalized End-to-End Loss for Speaker Verification. — 2020.

[27] Mathieu M. Couprie C., LeCun Y. Deep multi-scale video prediction beyond mean square error. // arXiv preprint arXiv:1511.05440. — 2015.

[28] Mehri S. Kumar K. Gulrajani I. Kumar R. Jain S. Sotelo J.-Courville A., Y. Bengio. Samplernn: An unconditional end-to-end neural audio generation model. // arXiv preprint arXiv:1612.07837. — 2016.

[29] Morise M. Yokomori F., Ozawa K. World: A vocoder-based high-quality speech synthesis system for real-time applications. // IEICE Transactions on Information and Systems. — 2016. — P. E99.D(7):1877–1884.

[30] Neekhara P. Donahue C. Puckette M. Dubnov S., McAuley J. Expediting tts synthesis with adversarial vocoding. // arXiv preprint arXiv:1904.07944. — 2019.

[31] Odena et al. Deconvolution and Checkerboard Artifacts. — 2016.

[32] Oord A. v. d. 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.

[33] Ping W. Peng K. Gibiansky A. Arik S. O. Kannan A. Narang-S. Raiman J., J. Miller. Deep voice 3: Scaling text-to-speech with convolutional sequence learning. // arXiv preprint arXiv:1710.07654. — 2017.

[34] Qiao Tian Yi Chen Zewang Zhang Heng Lu Linghui Chen Lei Xie-Shan Liu. TFGAN: Time and Frequency Domain Based Generative Adversarial Network for High-fidelity Speech Synthesis. — 2020.

[35] Ryuichi Yamamoto Eunwoo Song Jae-Min Kim. Parallel WaveGAN: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram. — 2020.

[36] 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. // IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). — 2018. — P. pp. 4779–4783.

[37] Sotelo J. Mehri S. Kumar-K. Santos J. F. Kastner K. Courville-A., Bengio Y. Char2wav: End-to-end speech synthesis. — 2017.

[38] Takuhiro Kaneko Hirokazu Kameoka Kou Tanaka Nobukatsu Hojo. CycleGAN-VC3: Examining and Improving CycleGAN-VCs for Mel-spectrogram Conversion. — 2020.

[39] Wang T.-C. Liu M.-Y. Zhu J.-Y. Liu G. Tao-A. Kautz J., Catanzaro B. Video-to-video synthesis. // Advances in Neural Information Processing Systems (NIPS). — 2018.

[40] Wang T.-C. Liu M.-Y. Zhu J.-Y. Tao A. Kautz-J., Catanzaro B. High-resolution image synthesis and semantic manipulation with conditional gans. // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. — 2018.

[41] Wang Y. Skerry-Ryan R.-Stanton D. Wu-Y. Weiss R. J.-Jaitly N. Yang Z. Xiao Y. Chen Z.-Bengio S. Tacotron: Towards end-to-end speech synthesis. // arXiv preprint arXiv:1703.10135. — 2017.

[42] Xudong Mao Qing Li Haoran Xie Raymond YK Lau Zhen Wang, Smolley Stephen Paul. Least squares generative adversarial networks. // Proceedings of the IEEE International Conference on Computer Vision. — 2017.

[43] Yamamoto R. Song E., Kim J.-M. Probability density distillation with

generative adversarial networks for high-quality parallel waveform generation. // arXiv preprint arXiv:1904.04472. — 2017.

[44] Zhu J.-Y. Park T. Isola P., Efros A. A. Unpaired image-to-image translation using cycleconsistent adversarial networks. // Proceedings of the IEEE international conference on computer vision. — 2017.

[45] et al. Qian. AUTOVC: Zero-Shot Voice Style Transfer with Only Autoencoder Loss. — 2019.