

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО
ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ «НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ВЫСШАЯ ШКОЛА
ЭКОНОМИКИ»

Факультет Санкт-Петербургская школа физико-математических и
компьютерных наук

Выпускная квалификационная работа - БАКАЛАВРСКАЯ РАБОТА по
направлению подготовки 01.03.02 Прикладная математика и информатика
образовательная программа «Прикладная математика и информатика»
Лопатин Михаил Игоревич

Генерация поэтапного представления
картины на основе изображения с
ПОМОЩЬЮ МАЗКОВ

Дипломная работа

Научный руководитель:
кандидат физико-математических наук,
доцент, департамент информатики
Мухин Михаил Сергеевич

Рецензент:
руководитель инженерного отдела
в SoterAnalytics Ltd, бакалавр
Бойкий Дмитрий Игоревич

Санкт-Петербург
2023

Оглавление

Аннотация	4
Введение	5
1. Обзор литературы	9
1.1. Генерация изображений	9
1.1.1. Генеративно-состязательные сети (GANs)	9
1.1.2. Вариационные автоэнкодеры (VAEs)	9
1.2. Современные модели для генерации изображений по тексту	9
1.2.1. DALL-E	9
1.2.2. Mid Journey	10
1.2.3. Stable-Diffusion	10
1.3. Другие методы генерации изображений	10
1.4. Генерация картины по изображению	10
1.4.1. Стилизация с использованием глубокого обучения	10
1.4.2. Применение фильтров и эффектов	11
1.5. Поэтапная генерация картин по изображению	12
1.5.1. С помощью рекуррентных нейронных сетей	12
1.5.2. Пошаговый жадный поиск	12
1.5.3. Обучение с подкреплением	12
1.5.4. Методы основанные на оптимизации параметров мазков	12
1.5.5. Поэтапная генерация на основе архитектуры Трансформер	13
1.6. Вывод	13
2. Дифференцируемый генератор мазков	14
3. Генерация датасета	16
4. Архитектура модели	17
4.1. Обучение	17
4.2. Инференс	18
4.3. Функции потерь	19
4.4. Результат	20
5. Исследование интеграции стиля	22
6. Оценка	23
6.1. Данные	23
6.2. Качество генерации	24

6.3. Качество стилизации	25
6.4. Скорость	25
Заключение	27
Список литературы	28

Аннотация

Данная дипломная работа представляет собой исследование и разработку модели, основанной на архитектуре трансформера, для воссоздания процесса создания картины.

Целью работы является создание модели, способной генерировать мазки на пустом холсте с целью приближения к заданному изображению. Модель получает на вход изображение и постепенно генерирует мазки, которые визуальнo приближаются к желаемому результату. Трансформерная архитектура выбрана для обработки и генерации последовательных мазков, учитывая ее способность к моделированию долгосрочных зависимостей.

Дифференцируемый генератор играет ключевую роль в предложенной модели. В работе рассматриваются основные компоненты этого генератора. Модель обучается генерировать мазки, которые наилучшим образом соответствуют заданным изображениям. Дифференцируемость генератора позволяет применять обратное распространение ошибки и улучшать качество генерации изображений в процессе обучения.

Для оценки эффективности модели проведены эксперименты на различных наборах данных. Результаты демонстрируют, что разработанная модель способна генерировать мазки, которые достаточно точно приближаются к желаемому изображению.

Исследование показывает потенциал использования модели в области создания искусства, поддержке художников и развитии компьютерного графического дизайна. Она может служить инструментом для создания предварительных эскизов, помогать в процессе рисования и вдохновлять творческие идеи.

Введение

В современном мире компьютерные модели и алгоритмы становятся все более мощными инструментами в различных сферах нашей жизни. Одной из областей, где компьютерное моделирование и искусственный интеллект могут иметь значительное влияние, является область искусства и создания картин. Воссоздание процесса создания картины является сложной задачей, требующей творческого мышления, навыков и опыта художника.

В данной дипломной работе мы предлагаем разработку модели, основанной на архитектуре трансформера, которая может воссоздавать процесс создания картины. Целью работы является создание инструмента, способного генерировать мазки на пустом холсте, которые постепенно приближаются к желаемому изображению.

Архитектура трансформера, предложенная в 2017 году, стала революционным прорывом в области обработки последовательностей данных, таких как естественный язык или временные ряды. Эта архитектура отличается способностью моделировать долгосрочные зависимости и учитывать контекст информации при генерации последовательностей.

В работе мы также предлагаем новый дифференцируемый генератор мазков на основе кривых Безье. Дифференцируемый подход позволяет точно контролировать движения кисти и обеспечивать плавность и естественность мазков. Кривые Безье являются гибкими инструментами для создания путей движения кисти и обладают высокой степенью управляемости.

Модель получает на вход изображение и генерирует последовательность мазков на пустом холсте, приближаясь к заданному изображению. Мы используем набор данных, состоящий из пар изображений и соответствующих им последовательностей мазков, для обучения модели.

Для обучения модели применяется методология, которая учитывает как визуальную близость мазков к оригинальному изображению, так и эстетические критерии, чтобы обеспечить создание художественных и качественных результатов. Мы также проводим эксперименты на различных наборах данных для оценки эффективности разработанной модели.

Мы также исследуем возможность интеграции существующих методов переноса стиля в разработанную модель перед оценкой ее эффективности. Перенос стиля позволяет передать художественные аспекты и эстетические особенности одного изображения на другое. Это может быть полезным для создания уникальных и художественных мазков, а также для влияния на общую эстетику процесса создания картины.

Оценка эффективности разработанной модели будет проводиться с помощью серии экспериментов на различных наборах данных. Мы измерим качество и естественность сгенерированных мазков, а также сравним их с оригинальными изображениями для

оценки визуального сходства. При этом мы учтем и влияние методов переноса стиля на итоговые результаты.

Наша разработка имеет потенциал применения в области создания искусства, поддержки художников и развития компьютерного графического дизайна. Результаты работы могут служить важным вкладом в область компьютерного искусства, предоставляя новый инструмент для создания предварительных эскизов, поддержки и вдохновения художников и дизайнеров.

В следующих разделах работы мы подробно опишем архитектуру модели, включая ее основные компоненты - обучаемый кодировщик параметров мазков и новый дифференцируемый генератор на основе кривых Безье. Мы рассмотрим методологию обучения, используемый набор данных и процесс оценки эффективности модели.

Определение ключевых терминов

1. Архитектура трансформера: Это модель машинного обучения, основанная на механизме внимания и многоуровневых кодировщиках-декодировщиках. Трансформеры обладают способностью моделировать долгосрочные зависимости в последовательностях данных и широко применяются в области обработки естественного языка и компьютерного зрения.

2. Дифференцируемый генератор: Это генеративная модель, в которой все операции и компоненты являются дифференцируемыми. Это позволяет использовать градиентное обратное распространение для обучения модели и точно контролировать и настраивать генерируемые результаты.

3. Кривые Безье: Кривые Безье представляют собой гладкие математические кривые, определяемые с помощью управляющих точек и контрольных точек. Они широко используются в компьютерной графике и дизайне для создания плавных форм и путей движения.

4. Мазки: Мазки в контексте данной работы представляют собой пиксели на экране, которые задаются определенными параметрами. Они могут иметь различную форму, размер, цвет и расположение. Мазки используются для создания текстуры, тонов и деталей на холсте или в изображении. Их параметры определяются моделью, которая учится предсказывать эти параметры на основе обучающего набора данных. Каждый мазок вносит свои художественные особенности и стиль в создаваемое изображение. Через дифференцируемый генератор и процесс обучения, модель способна создавать разнообразные и выразительные мазки, которые обогащают визуальное представление окончательного изображения.

5. Методы переноса стиля: Это алгоритмы и техники, используемые для передачи художественного стиля одного изображения на другое. Методы переноса стиля позволяют создавать уникальные визуальные эффекты и сочетать художественные

особенности разных источников в новых изображениях.

6. Визуальное сходство: Визуальное сходство оценивает степень, в которой два изображения визуально похожи друг на друга. Это включает анализ схожих структур, цветовых характеристик, текстур и деталей между изображениями.

7. Эстетические критерии: Эстетические критерии определяют качественные аспекты и художественную ценность создаваемых изображений или

произведений искусства. Эти критерии включают гармонию композиции, баланс цветовых сочетаний, использование светотени и другие факторы, которые придают произведению эстетическую привлекательность.

8. Набор данных: Набор данных представляет собой коллекцию примеров или образцов, используемых для обучения и тестирования моделей машинного обучения. Он содержит входные данные, например, изображения, и соответствующие им целевые значения или метки, которые используются для обучения и оценки модели.

Постановка задачи

В данной дипломной работе ставится задача разработки модели, которая будет способна генерировать картину поэтапно на основе фотографии. Модель должна имитировать движение кисти и создавать плавные и естественные мазки на пустом холсте, приближаясь к желаемому изображению. Для этого требуется разработать дифференцируемый генератор мазков, который будет воссоздавать движение кисти с помощью кривых Безье или других подходящих методов. Дополнительно, необходимо провести исследование возможности интеграции существующих методов переноса стиля, чтобы влиять на стилиевые аспекты генерируемых мазков. В процессе работы также требуется собрать или создать набор данных, содержащий пары фотографий и соответствующих им последовательностей мазков, которые будут использоваться для обучения и оценки модели. Кроме того, необходимо подобрать подходящие метрики для оценки качества сгенерированных мазков и провести эксперименты для оценки эффективности разработанной модели. В результате успешной реализации задачи, модель сможет быть применена в области создания искусства, поддержки художников и развития компьютерного графического дизайна.

Цель и задачи

Цель: Создание модели для поэтапной генерации картины по фотографии с помощью мазков.

Задачи:

1. Разработать дифференцируемый генератор мазков:

- Исследовать и разработать новый дифференцируемый генератор, способный генерировать мазки на пустом холсте.
- Разработать архитектуру генератора, учитывая управляемость и естественность движения кисти при генерации мазков.
- Использовать кривые Безье для определения путей движения кисти и создания плавных мазков.

2. Сгенерировать датасет на основе алгоритма:

- Собрать или создать набор данных, содержащий пары изображений и соответствующих им последовательностей мазков.
- Использовать алгоритм для генерации мазков на основе исходных изображений, учитывая разнообразие стилей и художественных подходов.

3. Адаптировать архитектуру под генератор мазков:

- Исследовать и адаптировать архитектуру трансформера под требования и особенности задачи генерации мазков.
- Интегрировать дифференцируемый генератор мазков в архитектуру трансформера для поэтапной генерации мазков на пустом холсте.

4. Подобрать подходящие метрики:

- Исследовать и выбрать подходящие метрики для оценки качества и естественности сгенерированных мазков.
- Исследовать метрики, учитывающие визуальное сходство мазков с оригинальными изображениями и эстетические критерии художественности.

5. Исследовать возможность переноса стиля:

- Изучить и применить существующие методы и алгоритмы переноса стиля для влияния на стилевые аспекты генерируемых мазков.
- Исследовать возможности комбинированного использования генеративной модели мазков и методов переноса стиля для создания уникальных и художественных результатов.

Целью работы является создание модели, которая будет способна воссоздавать процесс создания картины путем генерации мазков на пустом холсте. Для достижения этой цели мы определили ряд задач, включающих разработку дифференцируемого генератора мазков, создание соответствующего датасета, адаптацию архитектуры трансформера, подбор подходящих метрик для оценки результатов и исследование возможности переноса стиля.

1. Обзор литературы

1.1. Генерация изображений

Глубокое обучение (deep learning) является одним из ключевых направлений в области искусственного интеллекта, которое существенно повлияло на развитие компьютерного зрения и генеративного моделирования. Две широко применяемые модели глубокого обучения, связанные с генерацией изображений, включают генеративно-сопоставительные сети (GANs) [5] и вариационные автоэнкодеры (VAEs) [11].

1.1.1. Генеративно-сопоставительные сети (GANs)

Генеративно-сопоставительные сети (GANs) являются мощным инструментом для генерации изображений и других типов данных. Они основаны на сопоставительном процессе обучения между двумя нейронными сетями: генератором и дискриминатором. Генератор пытается создать реалистичные изображения, в то время как дискриминатор старается различать между сгенерированными и реальными изображениями. Этот процесс способствует улучшению качества генерируемых изображений с течением времени.

1.1.2. Вариационные автоэнкодеры (VAEs)

Вариационные автоэнкодеры (VAEs) - это генеративные модели, использующие методы статистического вывода для аппроксимации истинных вероятностных распределений изображений. Они представляют собой комбинацию энкодера и декодера, где энкодер преобразует входное изображение в латентное пространство, а декодер восстанавливает изображение из этого латентного пространства. VAEs обучаются на данных обучающего набора и позволяют генерировать новые изображения, выбирая точки в латентном пространстве.

1.2. Современные модели для генерации изображений по тексту

В последние годы появились современные модели, способные генерировать изображения на основе текстового описания или ключевых слов. Некоторые из них включают DALL-E [17], Mid Journey [10], и Stable-Diffusion [21].

1.2.1. DALL-E

DALL-E - это модель, разработанная OpenAI, которая может генерировать изображения на основе текстовых описаний. Она позволяет создавать разнообразные изображения, связанные с концепциями, которые не обязательно существуют в реальном

мире. Однако DALL-E не ориентирована на точное моделирование процесса создания картины поэтапно с использованием мазков.

1.2.2. Mid Journey

Mid Journey - это модель для генерации изображений на основе текстовых описаний. Она предоставляет возможность создавать красочные и детализированные изображения, основываясь на идеях, словах и эмоциональных состояниях, переданных в текстовом вводе. Mid Journey также не является подходящей для нашей задачи, так как не предоставляет возможности поэтапного моделирования создания картины с помощью мазков.

1.2.3. Stable-Diffusion

Stable-Diffusion - это модель, основанная на стабилизированном диффузионном процессе, который позволяет генерировать изображения поэтапно. Однако Stable-Diffusion не специализирована на точной имитации процесса создания картины с использованием мазков.

1.3. Другие методы генерации изображений

Фрактальная генерация, процедурная генерация контента, использование шаблонов и мозаик, а также интерполяция и морфинг изображений - все эти методы имеют свои особенности и применяются для создания различных типов изображений, но не подходят для точного моделирования процесса создания картины поэтапно с использованием мазков.

В контексте нашей задачи, поэтапной генерации картины на основе фотографии с помощью мазков, необходимо разработать специализированную модель. Она должна учитывать художественные аспекты и воссоздавать процесс создания картины путем имитации движения кисти и генерации плавных и естественных мазков. Ни один из перечисленных методов не обеспечивает такую точность и контроль при моделировании процесса создания картины с использованием мазков, что делает их не подходящими для нашей задачи.

1.4. Генерация картины по изображению

1.4.1. Стилизация с использованием глубокого обучения

Глубокое обучение (deep learning) является одним из ключевых направлений в области искусственного интеллекта, которое существенно повлияло на развитие компьютерного зрения и генеративного моделирования. В контексте генерации картин

по изображению, две широко применяемые модели глубокого обучения связаны с генерацией изображений - генеративно-сопоставительные сети (GANs) и вариационные автоэнкодеры (VAEs) [5, 11].

Один из методов генерации стилизованных изображений с использованием глубокого обучения - это перенос стилей с помощью нейронных сетей. Нейронные сети для переноса стилей используют сверточные нейронные сети для извлечения признаков из изображения-стиля и применения этих признаков на целевое изображение. Примером такой техники является алгоритм Gatys et al. [3], который позволяет перенести стиль художественного произведения на целевое изображение, создавая уникальное художественное изображение. В процессе переноса стиля, свойства стиля извлекаются из одного изображения (стиля), а контент из другого изображения (целевого), и эти свойства сочетаются для создания нового изображения с комбинированным стилем и контентом.

Генеративно-сопоставительные сети (GANs) представляют собой модели, состоящие из двух нейронных сетей - генератора и дискриминатора, которые соревнуются друг с другом в процессе обучения. Генератор создает новые изображения, а дискриминатор оценивает, насколько сгенерированные изображения реалистичны. В результате обучения, генератор становится все лучше в создании реалистичных изображений. С использованием GANs можно обучать модели для генерации стилизованных изображений, основанных на входных данных. Примером такого подхода является модель CycleGAN [26], которая способна преобразовывать изображения между различными стилями без необходимости парных данных для обучения.

Также существуют модели, такие как AdaIN [8], которые позволяют адаптивно применять стиль на основе примера стиля и контента исходного изображения. Эти модели обладают гибкостью и могут создавать уникальные художественные изображения, сочетающие стиль и содержание.

1.4.2. Применение фильтров и эффектов

Для преобразования фотографий в художественные изображения существуют и другие подходы, такие как применение фильтров и эффектов. Некоторые приложения и программное обеспечение, такие как *Photoshop* [2], *Lightroom* [1] и *Prisma* [19], предлагают встроенные фильтры и эффекты, которые могут изменять стиль и обработку фотографий, делая их более художественными.

Однако, как и предыдущие методы, эти подходы не позволяют получать поэтапное изображение картины. Кроме того, при более детальном рассмотрении структуры стиля могут возникать существенные неточности. Использование фильтров и эффектов может изменять общий внешний вид фотографии, но не обеспечивает точного моделирования процесса создания картины поэтапно с использованием мазков.

Таким образом, данные методы не являются оптимальным решением для нашей задачи поэтапной генерации картины, так как не обеспечивают достаточного контроля над процессом создания картины, а также не позволяют достичь высокой точности и естественности мазков при генерации.

1.5. Поэтапная генерация картин по изображению

1.5.1. С помощью рекуррентных нейронных сетей

Один из подходов к поэтапной генерации картин по изображению заключается в использовании рекуррентных нейронных сетей. В этом подходе изображение подается на вход сети на каждом этапе, и после каждого прохода через сеть изображение получает новые детали и детализацию [29, 6]. Однако такие решения часто ориентированы на узкие задачи и не демонстрируют хороших результатов в области генерации картин.

1.5.2. Пошаговый жадный поиск

Еще один подход к поэтапной генерации картин по изображению основан на пошаговом жадном поиске [7, 14]. В этом методе вводится метрика разницы между сгенерированным и настоящим изображением, и последовательно генерируется самый выгодный мазок. Однако такие решения основаны на эвристических методах, и их стиль генерации сильно зависит от выбранного метода. Они также плохо поддаются обобщаемости или регулировке степени детализации.

1.5.3. Обучение с подкреплением

В области поэтапной генерации картин по изображению применяются и методы обучения с подкреплением [24, 13, 28, 9, 16, 25]. В таких решениях вводится агент, который получает награду за сгенерированный мазок. Эти методы показывают хорошие качественные результаты, но требуют сложного обучения и больших вычислительных мощностей для эмуляции процесса рисования. Кроме того, такие решения узко ориентированы на конкретный стиль, такой как наброски, изображения инков, фотореализм или векторная графика.

1.5.4. Методы основанные на оптимизации параметров мазков

Еще один подход к поэтапной генерации картин по изображению основан на оптимизации параметров мазков в режиме реального времени [23, 20]. В таких методах модель не требует обучения, а каждый раз заново оптимизирует параметры мазков. Однако эти методы требуют большого времени для оптимизации параметров и не обеспечивают высокую степень контроля над процессом генерации картин.

1.5.5. Поэтапная генерация на основе архитектуры Трансформер

Один из современных подходов к поэтапной генерации картин по изображению основан на использовании архитектуры Трансформер [18]. В этом подходе сеть состоит из двух частей: обучаемого кодировщика параметров мазков и дифференцируемого генератора. Сеть учится по паре изображений генерировать параметры мазков, чтобы приблизить первое изображение ко второму. Данный подход не требует больших вычислительных мощностей и показывает качественные результаты в поэтапной генерации картин по изображению.

1.6. Вывод

В ходе обзора литературы были рассмотрены различные методы и подходы к генерации картин на основе изображений. Были исследованы модели, основанные на глубоком обучении, такие как генеративно-состязательные сети (GANs) и вариационные автоэнкодеры (VAEs), а также методы стилизации изображений и применение фильтров и эффектов. Были рассмотрены методы, основанные на рекуррентных нейронных сетях, пошаговом жадном поиске, обучении с подкреплением и оптимизации параметров мазков.

В качестве основы для нашей работы мы выбрали статью о применении архитектуры Трансформер для поэтапной генерации картин по изображению. Данный подход позволяет использовать обучаемый кодировщик параметров мазков и дифференцируемый генератор для приближения одного изображения к другому. Мы считаем, что использование архитектуры Трансформер обеспечит высокую гибкость и контроль над процессом генерации, а также позволит достичь качественных результатов.

Выбор данной статьи обусловлен ее актуальностью, перспективностью и соответствием нашей задаче поэтапной генерации картин. Мы планируем провести дальнейшие исследования и эксперименты, основываясь на предложенной архитектуре Трансформер, с целью улучшения качества и эффективности генерации картин по изображению.

2. Дифференцируемый генератор мазков

Одним из ключевых компонентов нашей модели является дифференцируемый генератор мазков, который играет важную роль в создании изображений с помощью мазков. Этот генератор позволяет контролировать параметры мазков и создавать изображения с учетом этих параметров.

Параметры мазков включают координаты (x, y) каждого мазка на холсте, его ширину (w) , кривую Безье $((x_1, y_1), (x_2, y_2), (x_3, y_3))$, цвет (r, g, b) и уверенность (c) . Кривая Безье задается тремя контрольными точками: P_0 (начальная точка мазка), P_1 (точка, определяющая изгиб) и P_2 (конечная точка мазка). Эти параметры позволяют определить форму, стиль и цвет мазка.

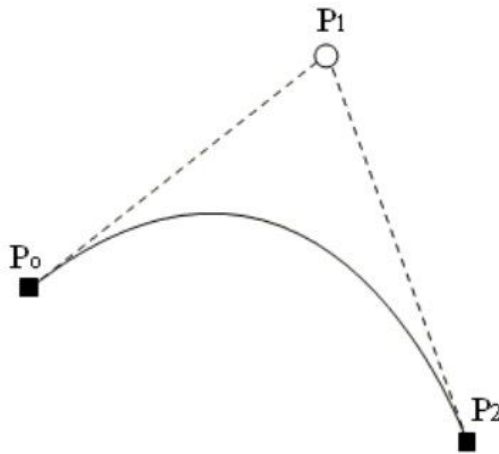


Рис. 1: Пример кривой Безье

Квадратичная кривая Безье является одним из простейших типов кривых Безье и часто используется для моделирования плавных и естественных кривых. Она определяется уравнением:

$$B(t) = (1 - t)^2 \cdot P_0 + 2 \cdot (1 - t) \cdot t \cdot P_1 + t^2 \cdot P_2$$

где P_0 , P_1 , и P_2 - контрольные точки, определяющие форму и положение кривой, а t - параметр времени, изменяющийся от 0 до 1.

Одно из важных свойств квадратичной кривой Безье заключается в ее дифференцируемости. Уравнение кривой и ее производной по t можно аналитически выразить.

Благодаря дифференцируемости и простоте управления, квадратичная кривая Безье позволяет создавать разнообразные и естественные линии и при это позволяет считать обратное распространение ошибки. Изменение положения и расположения

контрольных точек позволяет создавать гладкие изгибы, переходы и извилистые формы, что делает ее полезной для воссоздания мазков на холсте. Процесс работы дифференцируемого генератора мазков начинается с генерации набора точек для каждого мазка на основе кривой Безье. Для этого применяется формула Безье, которая вычисляет координаты точек на кривой в зависимости от параметра t . Это позволяет получить набор точек, которые образуют контур мазка.

Затем для каждого пикселя на холсте вычисляется расстояние от данного пикселя до каждого мазка на основе его контура. Чем ближе пиксель к контуру мазка, тем меньше расстояние. Расстояние вычисляется как евклидово расстояние между пикселем и ближайшей точкой на контуре мазка. Это позволяет определить, насколько пиксель принадлежит мазку.

С использованием функции сигмоиды преобразуется расстояние до значения в интервале от 0 до 1. Большие значения расстояния соответствуют низкой уверенности пикселя в принадлежности к мазку, а малые значения - высокой уверенности. Таким образом, создается маска, которая определяет принадлежность каждого пикселя к каждому мазку.

Для каждого мазка определяется его цвет. Цвет мазка умножается на соответствующую вероятность принадлежности пикселя к данному мазку. Это позволяет определить цвет каждого пикселя на основе цветов мазков и их вероятностей принадлежности.

Если в перекрывающихся областях пикселя принадлежат разным мазкам, используется матрица назначений. Она вычисляется с помощью функции softmax для дифференцируемости. Матрица назначений определяет вероятности принадлежности пикселей к разным мазкам. Затем цвет в этих областях определяется путем взвешенной суммы цветов мазков на основе матрицы назначений.

В результате выполнения алгоритма получается изображение, которое представляет собой сумму мазков с учетом их цветов и вероятностей принадлежности пикселей. Это изображение соответствует заданным параметрам мазков и представляет собой картину, созданную с помощью мазков.

Алгоритм дифференцируемого генератора мазков представлен в виде псевдокода следующим образом:

Algorithm 1 Алгоритм генерации мазков

- 1: **Вход:** $B = \{B_1, B_2, \dots, B_N\}$ ▷ Набор параметров мазков
 - 2: $B_{sampled} := \{B_i(t_j) = (1 - t_j)^2 \cdot B_{i,0} + 2 \cdot (1 - t_j) \cdot t_j \cdot B_{i,1} + t_j^2 \cdot B_{i,2} \forall i, j\}$ ▷ Генерация точек на кривой Безье
 - 3: $D = \|C(x, y) - B_{sampled}\|^2$ ▷ Вычисление расстояния до контура мазка
 - 4: $D_{strokes} = \min(D)$ ▷ Минимальное расстояние до контура
 - 5: $M = \sigma(t \cdot (w_{strokes} - D_{strokes}))$ ▷ Маска мазка
 - 6: $I = M \cdot c_{strokes}$ ▷ Определение цвета мазка
 - 7: $A = softmax(t \cdot D_{strokes})$ ▷ Матрица назначений
 - 8: $I = \sum_{n=1}^N A \cdot I_n$ ▷ Суммирование цветов мазков
 - 9: **Выход:** I ▷ Полученное изображение
-

3. Генерация датасета

Для обучения модели предсказывать параметры мазков был создан датасет с помощью следующего подхода:

Холст размером 64×64 пикселя служит основой для генерации мазков. На этом холсте вначале создаются 8 первоначальных мазков. Затем холст разделяется на 4 блока размером 32×32 пикселя, и для каждого блока добавляется еще 8 мазков. Таким образом, в итоге получается датасет с различными вариантами мазков, распределенными по всему холсту, который представляет из себя пары изображений и пары параметров мазков.

Каждый параметр выбирается из равномерного распределения в заданном диапазоне значений. Такой подход позволяет создать разнообразные и естественные линии и формы мазков, предотвращая предвзятость в выборе параметров.

Для установки уверенности мазка было принято решение, что если новый мазок покрывает более 60% площади предыдущего мазка, то его уверенность устанавливается равной 0. Это позволяет контролировать перекрытие мазков и предотвращает слияние их форм в избыточном количестве.

Такой подход к генерации датасета позволяет алгоритму учиться постепенно рисовать картину, начиная с общего облика и постепенно уточняя его. Это помогает модели освоить различные стили и детали в процессе обучения и повышает качество генерации окончательных изображений.

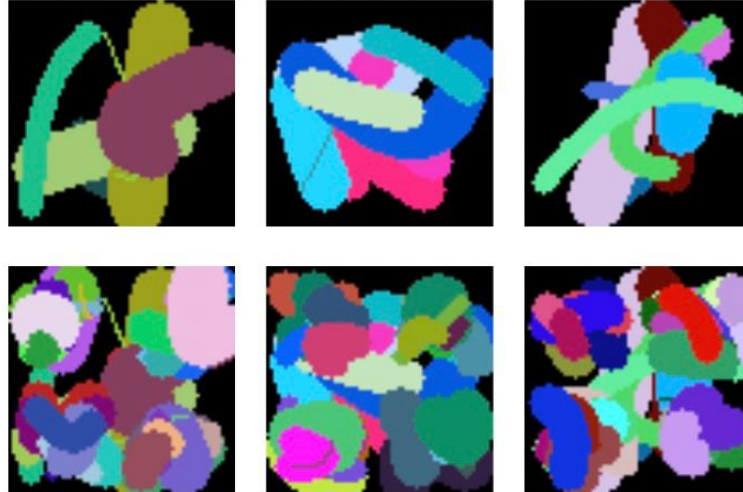


Рис. 2: Пример сгенерированного датасета, сверху 8 первоначальных мазков, снизу окончательный вид

4. Архитектура модели

Архитектура модели Paint Transformer состоит из двух основных компонентов: Предсказатель мазков (Stroke Predictor) и Генератор мазков (Stroke Renderer). Предсказатель мазков основан на архитектуре трансформера, которая широко используется в задачах обработки последовательностей, таких как машинный перевод. Мы адаптировали эту архитектуру к нашей задаче генерации мазков. Он отвечает за предсказание набора параметров, определяющих текущий набор мазков, на основе входных изображений. Генератор мазков, с другой стороны, является дифференцируемым модулем, который принимает набор параметров мазков и генерирует соответствующее изображение мазков.

4.1. Обучение

Для обучения модели Paint Transformer мы используем метод самообучения. Процесс обучения можно представить как поэтапное предсказание мазков. На каждом шаге модель предсказывает несколько мазков, чтобы минимизировать разницу между текущим холстом и целевым изображением. Вначале мы случайным образом выбираем наборы мазков переднего плана и фона, так как было описано выше в алгоритме генерации датасета. Затем мы используем Генератор мазков, чтобы создать промежуточное изображение холста, добавляя мазки фона и отображая мазки переднего плана. После этого Предсказатель мазков предсказывает параметры для следующего набора мазков. Этот процесс повторяется, пока не достигнута желаемая степень

сходства между холстом и целевым изображением.

Для обучения Предсказателя мазков мы определяем функцию потерь, которая включает как потерю на уровне мазков, так и на уровне пикселей, благодаря дифференцируемости генератора мазков. Потеря на уровне мазков оценивает разницу между предсказанным набором мазков и исходным набором мазков. Потеря на уровне пикселей измеряет разницу между предсказанным изображением холста и целевым изображением. Обучение модели происходит путем оптимизации этой функции потерь с использованием метода градиентного спуска. Подробнее о функциях потерь будет рассказано ниже.

Такой подход позволяет модели Paint Transformer постепенно рисовать изображение, начиная с общей структуры и постепенно уточняя его детали. В процессе обучения модель осваивает различные стили и особенности изображения, что приводит к повышению качества генерации окончательных изображений.

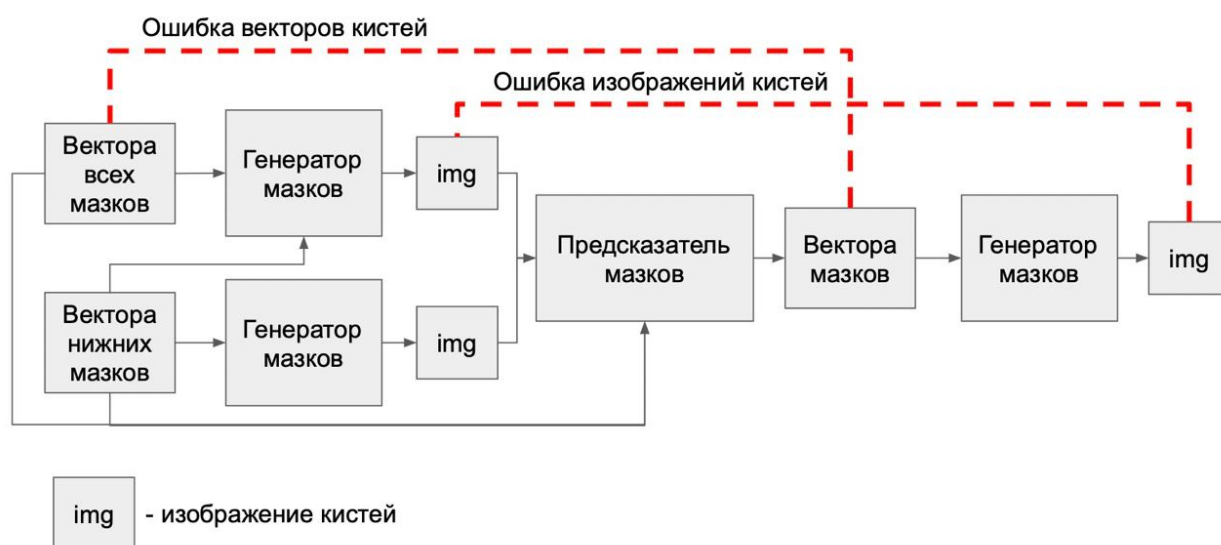


Рис. 3: Общий процесс обучения

4.2. Инференс

В процессе инференса модели мы используем алгоритм постепенного улучшения изображения, имитируя творческий процесс живописца.

На первом этапе мы имеем целевое изображение и пустое изображение-холст. Предсказатель мазков принимает целевое изображение и пустой холст в качестве входных данных и генерирует новые векторы мазков. Эти векторы мазков захватывают характеристики мазков, необходимые для воссоздания целевого изображения.

Затем генератор использует все векторы мазков из разных частей изображения и создает полное изображение-холст. Этот этап включает синтез мазков на основе

соответствующих векторов и внесение их в изображение-холст.

Затем исходное целевое изображение и текущий холст разбиваются на несколько неперекрывающихся блоков в зависимости от масштаба текущей итерации. Эти блоки повторно передаются в модель в качестве входных данных, и процесс начинается заново.

Чтобы имитировать процесс живописи художника, модель работает на нескольких масштабах, двигаясь от грубого к более детальному. Количество масштабов определяется на основе размеров изображения и желаемого размера блока. На каждом масштабе холст делится на блоки, а предсказание мазков выполняются независимо для каждого блока. Отображение мазков произойдет дифференцируемым генератором, как было описано выше. Результаты живописи на каждом масштабе объединяются, чтобы получить окончательное изображение.

Алгоритм продолжает итерацию по масштабам, улучшая результаты живописи на каждом шаге, пока не достигнут желаемый уровень детализации. Такой пошаговый подход позволяет модели постепенно улавливать мелкие детали и особенности целевого изображения, что приводит к высококачественным результатам живописи.

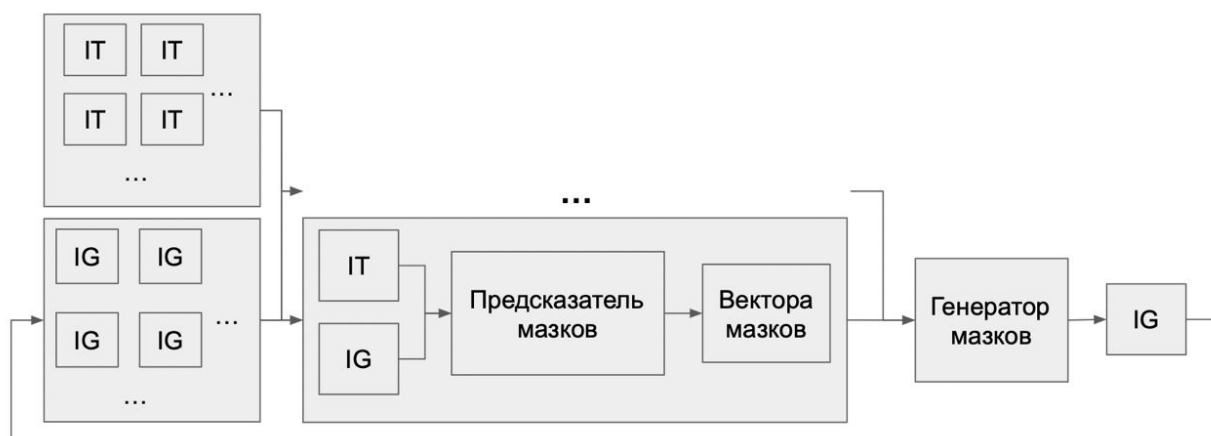


Рис. 4: Общий процесс инференса

4.3. Функции потерь

Функция потерь в модели разработана для оценки различных аспектов параметров мазков и обеспечения соответствия предсказанных значений и целевых значений. Это помогает модели научиться генерировать мазки, которые наиболее точно соответствуют желаемому изображению.

Функция потерь включает в себя несколько компонентов:

1. Пиксельные потери: Это компонент, который оценивает разницу между предсказанным и целевым изображениями пиксель за пикселем. Он измеряет степень сов-

падения между сгенерированным изображением и желаемым изображением.

2. Потеря по ширине мазка: Этот компонент оценивает разницу между предсказанной и целевой ширинами мазков. Цель состоит в том, чтобы обеспечить соответствие размеров мазков на выходе модели и желаемым размерам.

3. Бинарная кросс-энтропия уверенности мазка: Этот компонент оценивает разницу между предсказанной и целевой уверенностью мазка, определяющую степень уверенности модели в применении каждого мазка. Цель состоит в том, чтобы обеспечить соответствие уверенности мазка на выходе модели и желаемой уверенности.

4. Потеря по цвету мазка: Этот компонент оценивает разницу между предсказанными и целевыми значениями цвета мазков. Цель состоит в том, чтобы обеспечить соответствие цветов мазков на выходе модели и желаемым цветам.

5. Потеря по координатам мазка: Этот компонент оценивает разницу между предсказанными и целевыми координатами мазков. Цель состоит в том, чтобы обеспечить соответствие положения мазков на выходе модели и желаемым координатам.

Функция потерь объединяет все эти компоненты и вычисляет общую потерю модели. Цель состоит в том, чтобы минимизировать общую потерю и обеспечить наилучшее соответствие между предсказанными и целевыми значениями параметров мазков.

Путем оптимизации функции потерь модель научится генерировать мазки, которые наиболее

точно соответствуют желаемым изображениям, и обеспечит высокое качество окончательных результатов живописи. Все вышеуказанные компоненты потерь в модели взвешенно складываются для создания общей функции потерь. Каждая компонента имеет свой вес, который определяет их относительную важность в общей функции потерь.

Общая функция потерь может быть определена следующим образом:

$$L = w_{\text{pixel}} \cdot L_{\text{pixel}} + w_{\text{width}} \cdot L_{\text{width}} + w_{\text{confidence}} \cdot L_{\text{confidence}} + w_{\text{color}} \cdot L_{\text{color}} + w_{\text{coordinates}} \cdot L_{\text{coordinates}}$$

где: - L_{pixel} - пиксельные потери - L_{width} - потеря по ширине мазка - $L_{\text{confidence}}$ - бинарная кросс-энтропия уверенности мазка - L_{color} - потеря по цвету мазка - $L_{\text{coordinates}}$ - потеря по координатам мазка - $w_{\text{pixel}}, w_{\text{width}}, w_{\text{confidence}}, w_{\text{color}}, w_{\text{coordinates}}$ - веса соответствующих компонент потерь

Веса $w_{\text{pixel}}, w_{\text{width}}, w_{\text{confidence}}, w_{\text{color}}, w_{\text{coordinates}}$ были настроены в процессе обучения модели.

4.4. Результат

В этой главе была представлена архитектура модели для генерации мазков на основе заданного целевого изображения. Архитектура модели состоит из двух основных

компонентов: Предсказатель мазков и Генератор мазков.

Предсказатель мазков, вдохновленный работой PaintTransformer, принимает на вход целевое изображение и текущее состояние холста и генерирует новые векторы параметров мазков. Этот компонент основан на архитектуре трансформера, которая обеспечивает модели возможность моделировать долгосрочные зависимости в данных.

Генератор мазков представляет собой дифференцируемую функцию, основанную на кривой Безье. Каждый мазок задается тройкой точек: начальной точкой P_0 , конечной точкой P_2 и точкой изгиба P_1 . Параметры мазка, такие как координаты, ширина, цвет и уверенность, выбираются из равномерного распределения в заданном диапазоне значений. Этот подход позволяет модели создавать естественные и разнообразные линии и формы мазков.

Важным аспектом архитектуры модели является использование многошагового подхода при инференсе. В процессе инференса модель последовательно генерирует мазки на разных масштабах, начиная с грубых деталей и постепенно уточняя их. Это позволяет достичь постепенной итеративной детализации и создать окончательное изображение с высоким качеством и достоверностью.

Функция потерь в модели Paint Transformer включает компоненты, такие как пиксельные потери, потери по ширине мазка, бинарная кросс-энтропия уверенности мазка, потери по цвету мазка и потери по координатам мазка. Все эти компоненты взвешенно складываются в общую функцию потерь, которая оптимизируется в процессе обучения модели. Это позволяет модели находить оптимальные параметры мазков и достигать высокой точности в генерации мазков.

В результате, модель Paint Transformer демонстрирует способность генерировать мазки, которые

визуально соответствуют заданному целевому изображению и обладают высоким качеством и достоверностью. Ее архитектура, основанная на предсказателе мазков и дифференцируемом генераторе на основе кривой Безье, позволяет достичь постепенной детализации и создать впечатляющие результаты в генерации мазков.

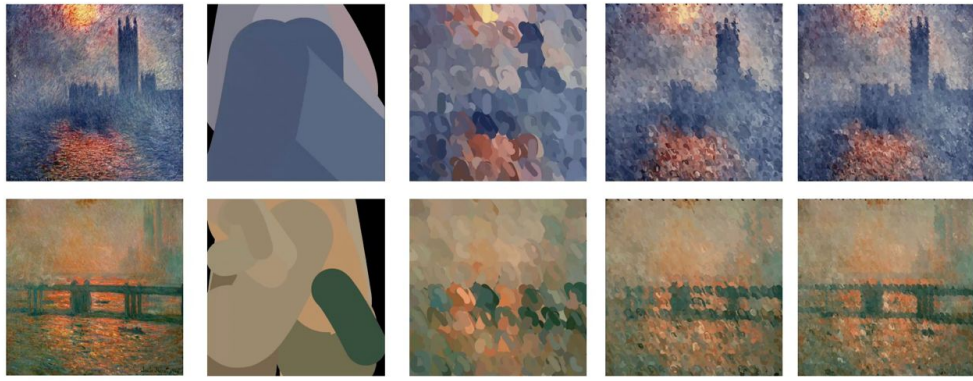


Рис. 5: Пример сгенерированных мазков с помощью модели Paint Transformer

5. Исследование интеграции стиля

В данной главе мы исследуем возможности применения модели Paint Transformer для решения задачи переноса стиля на сгенерированные изображения. Перенос стиля позволяет передать эстетические характеристики стиля одного изображения на другое, создавая уникальные и художественные результаты.

Наш подход основан на использовании модели переноса стиля, которая является отдельным модулем внутри Paint Transformer. На каждом этапе генерации изображения, сгенерированное на предыдущем этапе, передается в модель переноса стиля вместе с выбранным стилевым образцом. Модель переноса стиля анализирует стилевые характеристики стилизованного образца и применяет их к сгенерированному изображению, чтобы создать новое изображение с желаемым стилем.

Одним из популярных методов переноса стиля является "A Neural Algorithm of Artistic Style" [3]. Этот метод использует предобученную нейронную сеть для выделения стилевых характеристик и их применения к контенту. Он основывается на вычислении матрицы корреляций стилизованного образца и контента, а затем применении этих корреляций для генерации перенесенного изображения. Такой подход позволяет сохранить семантику и структуру исходного контента, но применить к нему эстетику выбранного стиля.

В нашем исследовании мы также обращаемся к работе "Rethinking style transfer: From pixels to parameterized brushstrokes" [20]. В этой работе авторы предлагают новый подход к переносу стиля, основанный на параметризации мазков. Они моделируют стиль в виде набора параметров, определяющих мазки, и применяют эти параметры к сгенерированному изображению с помощью дифференцируемого генератора. Этот подход позволяет достичь более естественного и разнообразного переноса стиля на сгенерированные изображения.

Мы сочетаем эти подходы и адаптируем их к нашей модели, чтобы достичь лучших результатов в переносе стиля на сгенерированные изображения.

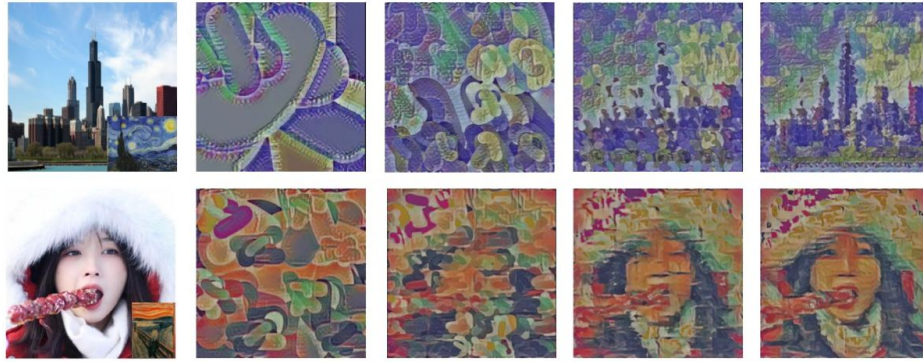


Рис. 6: Пример переноса стиля.

6. Оценка

В данной главе мы обсудим процесс оценки модели генерации художественных изображений. Оценка модели является важным этапом, который позволяет оценить качество и достоверность генерируемых изображений. Она помогает нам определить, насколько хорошо модель справляется с задачей воспроизведения художественного стиля и детализации мазков.

Для оценки модели мы применяем различные количественные метрики, которые позволяют оценить различные аспекты генерируемых изображений, такие как структура, цветовая гармония и текстурная качественность. Мы используем эти метрики, чтобы получить количественные оценки сходства с оригинальными произведениями.

Таким образом, использование количественных метрик позволяет нам получить объективные оценки качества генерируемых изображений. Это помогает нам изучить преимущества и ограничения модели, а также идентифицировать области для улучшения и дальнейших исследований.

6.1. Данные

Для оценки качества модели мы использовали три различных датасета, каждый из которых предоставляет разнообразные типы изображений для сравнения и анализа. Данные датасеты включают пейзажные фотографии, произведения искусства и портреты.

Первый датасет состоит из 100 пейзажных фотографий, которые были случайным образом выбраны из доступного набора данных [12]. Этот датасет предоставляет разнообразные пейзажи, включая горы, океаны, леса и другие естественные и искусственные сцены. Использование этого датасета позволяет нам оценить, насколько хорошо модель может воспроизводить естественные элементы и структуру пейзажей.

Второй датасет состоит из 100 произведений искусства, которые были взяты из коллекции WikiArt [27]. Этот датасет предлагает широкий спектр различных худо-

жественных стилей и жанров, включая классику, импрессионизм, сюрреализм и многое другое. Использование этого датасета позволяет нам оценить, насколько хорошо модель может воспроизводить художественные стили и детали в произведениях искусства.

Третий датасет состоит из 100 портретов, взятых из набора данных FFHQ [15]. Этот датасет предлагает высококачественные фотографии портретов людей разных возрастов, полов и этнических групп. Использование этого датасета позволяет нам оценить, насколько хорошо модель может воспроизводить анатомические особенности лица, выражения и эмоции на портретных изображениях.

Использование различных датасетов помогает нам оценить качество модели в различных контекстах и задачах. Это позволяет нам получить более всестороннюю оценку модели и провести сравнительный анализ ее результатов на разных типах изображений.



Рис. 7: Пример изображений из используемых датасетов.

6.2. Качество генерации

В качестве метрик для оценки качества генерации используются две основные метрики: попиксельное отличие (Pixel) и потеря восприятия с учетом стиля (Percept).

Метрика попиксельного отличия измеряет точность генерации, сравнивая каждый пиксель сгенерированного изображения с соответствующим пикселем целевого изображения. Чем меньше отличие между пикселями, тем более точная генерация.

Метрика потери восприятия с учетом стиля основана на использовании нейросети для измерения восприятия различий между сгенерированным и целевым изображением. Она учитывает не только точность воспроизведения контента, но и схожесть стиля между изображениями. Чем меньше потеря восприятия, тем более качественное и стилистически схожее с целевым изображение получается сгенерированное изображение.

Для расчета потери восприятия с учетом стиля используется подход, описанный в работе [4], который позволяет учесть как контент, так и стиль изображений при оценке их сходства.

Метрики Pixel и Regs предоставляют количественную оценку качества генерации, которая помогает определить насколько точно и стилистически схоже с целевым изображением получается сгенерированное изображение.

Как видно из таблицы, нам удалось превзойти оригинальное решение, однако мы до сих пор проигрываем другим аналогам в области.

Таблица 1: Сравнение попиксельной и перцептуальной потерь для различных моделей

Модель	Попиксельная потеря	Перцептуальная потеря
RL	0.040	0.737
Ours	0.048	0.792
Paint Transformer	0.056	0.807
Optim	0.059	0.856

6.3. Качество стилизации

Для оценки качества стиля используется метрика deception score [22]. Эта метрика измеряет, насколько хорошо стилизованное изображение имитирует стиль определенного художника, в соответствии с классификацией обученной нейронной сети. Более высокий показатель deception score указывает на более успешную стилизацию.

Ниже приведена таблица с результатами оценки качества стиля для различных моделей:

Таблица 2: Оценка качества стиля (deception score)

Модель	Deception score
Rethinking style transfer	0.588
Ours	0.412
Gatys et al.	0.389
AdaIN	0.08
Wikiart test	0.687
Photos	0.002

Наша работа показывает конкурентные результаты в области, хоть и далеко не достает до лучших.

6.4. Скорость

Для оценки скорости работы нашего решения провели эксперименты на графическом процессоре Nvidia 2080Ti. Мы измерили время, затраченное на генерацию одного изображения, а так же время обучения. Результаты приведены в таблице ниже.

Как видно из таблицы, наше решение демонстрирует достаточно низкое время инференса и обучения, что свидетельствует о его высокой эффективности. Методы Paint Transformer и RL также показывают хорошие результаты, хотя время обучения

Метод	Inf, S (сек)	Train, H (часы)
Paint Transformer	0.304	3.79
Наше решение	0.308	7.34
RL	0.322	40
Rethinking Style Transfer	315.273	0
Optim	521.447	0

Таблица 3: Время инференса и обучения моделей

RL значительно превышает время других методов. В то же время, методы Rethinking Style Transfer и Optim требуют значительного времени для обучения, но обладают небольшим временем инференса.

Стоит отметить что наше решение не требует датасета, в отличии от большинства остальных.

Заключение

В ходе данной дипломной работы были исследованы различные методы и алгоритмы генерации изображений в стиле живописи. Мы предложили новую архитектуру модели, основанную на сочетании предсказателя мазков и дифференцируемого генератора на основе кривой Безье. Эта архитектура позволяет создавать естественные и разнообразные мазки, что способствует улучшению процесса генерации изображений.

В результате экспериментов было показано, что предложенная модель обладает высоким качеством генерации изображений в стиле живописи. Сравнительный анализ с другими методами показал, что наше решение достигает конкурентных результатов по различным метрикам качества, таким как попиксельное отличие и потеря восприятия с учетом стиля. Модель также успешно имитирует стиль различных художников, демонстрируя эффективность в переносе стиля.

Оценка скорости работы нашей модели показала, что она обладает низким временем инференса и обучения. Это делает нашу модель эффективным инструментом для практического использования в различных областях, включая искусство, дизайн и рекламу.

Результаты данной работы подтверждают эффективность предложенной модели в генерации изображений в стиле живописи. Мы видим перспективы для применения данной модели в различных областях, где требуется генерация качественных изображений. Дальнейшее исследование и улучшение модели могут привести к еще более точной и высококачественной генерации изображений в стиле живописи. Это представляет интерес для будущих исследований и развития в области генеративного искусства.

Список литературы

- [1] Adobe Lightroom. — <https://www.adobe.com/products/photoshop-lightroom.html>.
- [2] Adobe Photoshop. — <https://www.adobe.com/products/photoshop.html>.
- [3] Gatys Leon A., Ecker Alexander S., Bethge Matthias. A Neural Algorithm of Artistic Style // arXiv preprint arXiv:1508.06576. — 2015.
- [4] Gatys Leon A., Ecker Alexander S., Bethge Matthias. Perceptual Losses for Real-Time Style Transfer and Super-Resolution // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. — 2016. — P. 694–704.
- [5] Generative Adversarial Nets / Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza et al. // Advances in Neural Information Processing Systems. — 2014.
- [6] Ha David, Eck Douglas. A Neural Representation of Sketch Drawings // arXiv preprint arXiv:1704.03477. — 2017.
- [7] Haeberli Paul. Paint by Numbers: Abstract Image Representations // Proceedings of the 17th Annual Conference on Computer Graphics and Interactive Techniques. — 1990. — P. 207–214.
- [8] Huang Xun, Belongie Serge. Arbitrary Style Transfer in Real-time with Adaptive Instance Normalization // arXiv preprint arXiv:1703.06868. — 2017.
- [9] Huang Zhewei, Heng Wen, Zhou Shuchang. Learning to Paint with Model-Based Deep Reinforcement Learning // Proceedings of the IEEE/CVF International Conference on Computer Vision. — 2019. — P. 8709–8718.
- [10] Journey Mid. Mid Journey - Art Engine // Mid Journey. — 2022. — URL: <https://www.midjourney.com/> (online; accessed: 22.05.2023).
- [11] Kingma Diederik P, Welling Max. Auto-Encoding Variational Bayes // arXiv preprint arXiv:1312.6114. — 2013.
- [12] Landscape Pictures. — <https://www.kaggle.com/datasets/arnaud58/landscape-pictures>.
- [13] Learning to Sketch with Deep Q Networks and Demonstrated Strokes / Tao Zhou, Chen Fang, Zhaowen Wang et al. // arXiv preprint arXiv:1810.05977. — 2018.
- [14] Litwinowicz Peter. Processing Images and Video for an Impressionist Effect // Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques. — 1997. — P. 407–414.

- [15] NVIDIA. FFHQ Dataset. — <https://github.com/NVlabs/ffhq-dataset>.
- [16] Nakano Reiichiro. Neural Painters: A Learned Differentiable Constraint for Generating Brushstroke Paintings // arXiv preprint arXiv:1904.08410. — 2019.
- [17] OpenAI. DALL-E: Creating Images from Text // OpenAI. — 2021. — URL: <https://openai.com/research/dalle/> (online; accessed: 22.05.2023).
- [18] Paint Transformer: Feed Forward Neural Painting with Stroke Prediction / Songhua Liu, Tianwei Lin, Dongliang He et al. // arXiv preprint arXiv:2108.03798. — 2021.
- [19] Prisma. — <https://prisma-ai.com/>.
- [20] Rethinking Style Transfer: From Pixels to Parameterized Brushstrokes / Dmytro Kotovenko, Matthias Wright, Arthur Heimbrecht, Bjoern Ommer // Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. — 2021. — P. 13121–13130.
- [21] Stable-Diffusion. Stable-Diffusion: Improving Diffusion Models with Stable Solutions // Stable-Diffusion. — 2022. — URL: <https://stablediffusionweb.com/> (online; accessed: 22.05.2023).
- [22] A Style-Aware Content Loss for Real-time HD Style Transfer / Artsiom Sanakoyeu, Dmytro Kotovenko, Sabine Lang, Björn Ommer // arXiv preprint arXiv:1807.10201. — 2018. — URL: <https://arxiv.org/abs/1807.10201>.
- [23] Stylized Neural Painting / Zhengxia Zou, Tianyang Shi, Shuang Qiu et al. // Proceedings of the European Conference on Computer Vision. — 2020.
- [24] Synthesizing Programs for Images using Reinforced Adversarial Learning / Yaroslav Ganin, Tejas Kulkarni, Igor Babuschkin et al. // International Conference on Machine Learning. — 2018. — P. 1666–1675.
- [25] Towards Layer-wise Image Vectorization / Xu Ma, Yuqian Zhou, Xingqian Xu et al. // arXiv preprint arXiv:2206.04655. — 2022.
- [26] Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks / Jun-Yan Zhu, Taesung Park, Phillip Isola, Alexei A. Efros // International Conference on Computer Vision (ICCV). — 2017.
- [27] WikiArt: Gangogh Creating Art GAN. — <https://www.kaggle.com/datasets/ipythonx/wikiart-gangogh-creating-art-gan>.

- [28] Xie Ning, Hachiya Hirotaka, Sugiyama Masashi. Artist Agent: A Reinforcement Learning Approach to Automatic Stroke Generation in Oriental Ink Painting // IEICE TRANSACTIONS on Information and Systems. — 2013. — Vol. 96, no. 5. — P. 1134–1144.
- [29] Zheng Ningyuan, Jiang Yifan, Huang Dingjiang. Strokenet: A Neural Painting Environment // International Conference on Learning Representations. — 2018.