

Расширение возможностей систем генерации изображений путем использования нейронных сетей

Д. Е. Мулявин, Р. В. Мальчева, А. А. Койбаш

E-mail: vertik555@mail.ru, raisa.malchea@yandex.ru, mr.koibash@yandex.ru

Аннотация:

Статья представляет обзор основных проблем, с которыми сталкиваются исследователи и практики при генерации изображений с использованием нейронных сетей. Рассматриваются ключевые аспекты, такие как вычислительная сложность и качество сгенерированных изображений. Предлагаются потенциальные решения для данных проблем, включая оптимизацию архитектур нейронных сетей, применение методов оптимизации и использование специализированных аппаратных ускорителей. Подводятся итоги перспектив развития исследований в данной области, а также указываются направления для будущих исследований и инноваций.

Введение

Генерация изображений, особенно в реальном масштабе времени, требует значительных временных затрат. Для повышения производительности систем генерации изображения их реализуют на графических процессорах, таких как платформа CUDA [1], или разрабатывают специализированные устройства [2]. При этом существует значительный объем типовых изображений (ландшафты, городские пейзажи и др.), накопленный за более чем 60 лет существования компьютерной графики. Поэтому генерировать подобные изображения традиционными методами не всегда оправдано. Исходя из этого, авторами предпринята попытка расширения возможностей систем генерации изображений путём применения искусственного интеллекта (ИИ).

Современные достижения в области ИИ стали ключевым фактором в развитии методов генерации изображений, применяемых в различных приложениях, начиная от компьютерной графики и заканчивая медицинскими исследованиями. Не смотря на значительные технологические прорывы, встречаемые в данной области, существует ряд программных и аппаратных проблем, которые препятствуют эффективной генерации качественных изображений.

Обзор основных аспектов генерации изображений с использованием нейронных сетей

При генерации изображений с использованием нейронных сетей следует учитывать несколько ключевых аспектов [3]:

- *архитектура нейронной сети.* Выбор подходящей архитектуры сети зависит от

конкретной задачи. Например, генеративно-состязательные сети (GAN), автокодировщики (AE), вариационные автокодировщики (VAE) и условные генеративные модели (CGAN) имеют различные свойства и применения;

- *функция потерь (Loss Function).* Это критерий, который определяет, насколько хорошо сгенерированные изображения соответствуют оригинальным данным. В зависимости от архитектуры сети и поставленных целей могут применяться различные функции потерь, такие как бинарная кросс-энтропия, среднеквадратичная ошибка или перцептивные потери;

- *набор данных (Dataset).* Качество и разнообразие данных напрямую влияют на качество и разнообразие сгенерированных изображений. Необходимо выбирать или создавать набор данных, который соответствует целям генерации и содержит достаточно разнообразных примеров;

- *предобработка данных (Data Preprocessing).* Перед обучением модели данные обычно требуют предварительной обработки, такой как нормализация, изменение размера или аугментация, чтобы улучшить процесс обучения и качество результатов;

- *тренировка модели (Model Training).* Важно правильно настроить процесс обучения, включая выбор оптимизатора, скорости обучения, количество эпох и размер мини-пакета, чтобы достичь оптимальных результатов;

- *регуляризация и предотвращение переобучения (Regularization and Overfitting Prevention).*

Перечень проблем, возникающих в процессе генерации изображений с использованием методов искусственного интеллекта, включает следующие аспекты:

Одной из основных программных проблем является обеспечение высокого качества и реалистичности генерируемых изображений. Эта проблема обусловлена существующими ограничениями, связанными с точностью и детализацией создаваемых изображений.

Также проблемой остается скорость и эффективность алгоритмов. Процесс генерации изображений может быть крайне ресурсоемким и требовать значительного времени. Поэтому одной из задач является оптимизация алгоритмов для повышения скорости работы и эффективного использования вычислительных ресурсов [4].

Рассмотрим аппаратные проблемы.

Ограничения вычислительных ресурсов: Генерация изображений с использованием методов искусственного интеллекта может требовать значительных вычислительных мощностей, что делает этот процесс трудоемким и затратным с точки зрения аппаратного обеспечения.

Энергопотребление: Поскольку некоторые алгоритмы генерации изображений могут быть вычислительно интенсивными, они могут потреблять большие объемы энергии, особенно при работе на портативных устройствах, что представляет проблему для продолжительного использования и мобильных приложений [5].

Анализ проблем и их решение

Рассмотрим каждую из приведенных проблем и предложим решения, составленные на основе последних исследований.

Одной из главных проблем является качество и разнообразие *обучающих данных*. Модели машинного обучения, включая генераторы изображений, зависят от данных, на которых они обучаются. Если эти данные смещены, неполны или низкого качества, это отразится на результатах. Для решения этой проблемы самым оптимальным решением можно назвать улучшение качества и разнообразия наборов данных. Необходимо значительно расширять датасеты, из которых нейросеть сможет брать основу. Эта основа должна включать в себя большое количество разнообразного материала с высокими параметрами качества. Также можно использовать технику аугментации данных для увеличения количества и разнообразия обучающих примеров [6].

Следующей проблемой является неправильный *выбор архитектуры нейронной сети*. Архитектура обучающей модели играет ключевую роль в процессе генерации изображений. Она определяет структуру и функциональные возможности модели, влияя на ее способность извлекать признаки из входных данных и создавать новые изображения. В

контексте генерации изображений, архитектура обучающей модели определяет способность модели к адаптации к различным типам данных, реализации сложных зависимостей между пикселями изображения, а также способы управления качеством и разнообразием создаваемых изображений. В качестве возможного решения можно предложить применение новых архитектур нейронных сетей, которые лучше подходят для генерации изображений, например, GAN (Generative Adversarial Networks). Генеративно-сопоставительные модели представляют собой архитектуру, в которой две нейронные сети, генератор и дискриминатор, соревнуются между собой. Генератор создает изображения, а дискриминатор пытается различать реальные изображения от сгенерированных. Такая архитектура позволяет генератору научиться создавать реалистичные изображения, обманывая дискриминатор [4].

Следующей проблемой является задача оптимизации и переобучения. Эта задача является ключевой при обучении нейронных сетей для генерации изображений. Переобучение происходит, когда модель слишком сильно адаптируется к обучающим данным и начинает выделять ненужные шумы или детали, что приводит к ухудшению обобщающей способности модели на новых данных.

Для решения проблемы переобучения и оптимизации можно использовать следующие техники:

- **Регуляризация.** Техники регуляризации, такие как L1 и L2 регуляризация, добавляют штраф к функции потерь модели за большие веса или сложные зависимости между параметрами модели. Это помогает предотвратить избыточную сложность модели и уменьшить вероятность переобучения.

- **Использование дропаута.** Дропаут - это техника регуляризации, при которой случайно выбранные узлы нейронной сети игнорируются в процессе обучения. Это помогает предотвратить слишком сильную адаптацию нейронной сети к обучающим данным и уменьшить переобучение [7].

- **Ранняя остановка (Early stopping).** Ранняя остановка - это метод, при котором обучение модели прекращается, когда производительность модели на валидационном наборе данных начинает ухудшаться после достижения определенного максимума. Это позволяет избежать переобучения и сохранить обобщающую способность модели.

- **Использование аугментации данных.** Это помогает расширить обучающий набор данных и улучшить обобщающую способность модели, уменьшая вероятность переобучения [8].

Применение данных техник регуляризации и ранней остановки во время обучения нейронных сетей для генерации изображений может значительно улучшить качество и обобщающую способность моделей, а также ускорить их работу, предотвращая их переобучение и обеспечивая более стабильный процесс обучения.

Далее следует проблема, находящаяся на аппаратном уровне – недостаток быстродействия вычислительных устройств.

Проблема ограничения вычислительной мощности при генерации изображений с помощью нейронных сетей возникает из-за требовательности таких моделей к вычислительным ресурсам. Более сложные архитектуры и большие наборы данных могут потребовать значительного объема вычислительных ресурсов, включая высокопроизводительные графические процессоры (GPU) или тензорные процессоры (TPU), для обучения.

Для решения этой проблемы можно применить несколько подходов:

- Оптимизация архитектуры модели:

Использование менее ресурсоемких архитектур моделей может значительно снизить требования к вычислительной мощности. Например, использование Lightweight GAN или MobileNet для генерации изображений может помочь уменьшить нагрузку на вычислительные ресурсы без значительной потери качества.

- Дистилляция модели: Процесс дистилляции позволяет создавать более легкие и менее ресурсоемкие модели, сохраняя при этом их эффективность. Это достигается за счет передачи знаний более сложной модели на менее ресурсоемкую.

- Прогрессивное обучение: Можно начать обучение модели с более легкой и меньшей модели, а затем постепенно увеличивать ее размер и сложность по мере накопления вычислительных ресурсов или наличия данных.

- Использование облачных вычислений: Облачные сервисы предоставляют доступ к высокопроизводительным вычислительным ресурсам по запросу, что позволяет обучать и использовать модели нейронных сетей даже при ограниченной локальной вычислительной мощности.

- Применение аппроксимации моделей: Можно использовать методы аппроксимации, такие как квантизация, сжатие или обрезка моделей, чтобы уменьшить их размер и вычислительную сложность.

- Распределенное обучение: Распределение обучения модели между несколькими вычислительными устройствами или серверами может значительно ускорить процесс обучения [9, 10].

Проблема высокого энергопотребления при генерации изображений с помощью нейронных сетей схожа с предыдущей и возникает из-за высокой вычислительной нагрузки, которую они могут представлять. Более сложные модели, такие как GAN или большие наборы данных, требуют значительных вычислительных ресурсов, что приводит к увеличению энергопотребления. Это может стать проблемой с точки зрения экологической устойчивости и затрат на вычисления.

Чтобы решить эту проблему, можно применить несколько подходов:

- Оптимизация архитектуры модели:

Использование более легких и эффективных архитектур нейронных сетей может снизить энергопотребление. Например, использование MobileNet или EfficientNet для генерации изображений может уменьшить количество вычислительных операций и, следовательно, потребление энергии.

- Использование специализированных аппаратных решений: Применение специализированных аппаратных ускорителей, таких как тензорные процессоры (TPU) или ускорители глубокого обучения, спроектированных специально для работы с нейронными сетями, может существенно снизить энергопотребление при выполнении вычислений [11].

Примером реального решения является использование специализированных аппаратных ускорителей для выполнения вычислений нейронных сетей. Например, Google применяет свои тензорные процессоры (TPU) для выполнения операций нейронных сетей с высокой энергоэффективностью. Это позволяет сократить энергопотребление при обучении и инференсе моделей [12].

Современные системы с использованием искусственного интеллекта

Говоря о таких системах, нельзя обойти стороной технологии самой передовой компании в области графики – NVIDIA. Их графические адаптеры являются на сегодняшний день самыми популярными и совершенными. Во многом это тесно связано с областью кинематографа и видеоигр, поскольку именно они являются основным двигателем прогресса в области графики. Одними из важнейших для видеоигр на сегодня является технология RTX (Ray Tracing eXtreme) в связке с Deep Learning Super Sampling (DLSS), позволяющие создать фотореалистичную графику во многих современных играх.

Технология RTX является реализацией аппаратного ускорения трассировки лучей, основанного на интеграции тензорных и RT-ядер

в архитектуру графических процессоров, начиная с NVIDIA Turing. Основное новшество RTX — добавление RT-ядер для выполнения вычислений по трассировке лучей и тензорных ядер, предназначенных для ускоренной работы алгоритмов глубокого обучения, таких как DLSS.

RT-ядра — это специальные вычислительные модули в GPU, которые обрабатывают лучи для получения изображений с высокой реалистичностью, имитируя физическое поведение света. RT-ядра выполняют три основные функции: проверку пересечений, расчет теней и отражений, и вычисление взаимодействий с поверхностями. В ходе работы RT-ядра взаимодействуют с общими ядрами CUDA, которые обеспечивают основные графические вычисления. Каждое RT-ядро способно выполнять несколько операций пересечения за цикл, что существенно ускоряет рендеринг, особенно при высоких уровнях детализации и сложности сцен [13].

Тензорные ядра оптимизированы для матричных вычислений, которые играют ключевую роль в алгоритмах машинного обучения. В рамках RTX они выполняют операции свертки и матричные умножения, что позволяет ускорить модели глубокого обучения. Тензорные ядра предоставляют ресурсы для выполнения операций с плавающей запятой низкой точности (например, float 16 и float 8), что снижает нагрузку на вычислительные блоки и повышает производительность без заметных потерь в качестве изображения [13].

RT-ядра и тензорные ядра взаимодействуют с основными CUDA-ядрами, разделяя задачи по обработке графики и алгоритмов машинного обучения. CUDA-ядра обрабатывают геометрию, растеризацию и другие графические задачи, тогда как RT-ядра занимаются трассировкой лучей, а тензорные ядра поддерживают алгоритмы ИИ для оптимизации изображения (например, DLSS). Этот многозадачный подход позволяет эффективно распределять вычислительные ресурсы и минимизировать задержки в обработке графических сцен.

DLSS — это алгоритм сверхвысокого разрешения, который использует возможности глубокого обучения для повышения качества изображения при рендеринге с низким разрешением, уменьшая нагрузку на GPU. Его основой является нейронная сеть, обученная на большом количестве кадров высокого качества, что позволяет системе предсказывать и восстанавливать детали, не находящиеся в исходном изображении. DLSS состоит из двух основных компонентов: нейронной сети и механизма «апскейлинга». Нейронная сеть работает на тензорных ядрах, обучаясь на

заранее собранных наборах данных. Этот процесс включает в себя анализ кадров с низким разрешением и их улучшение до высокого разрешения с сохранением четкости и точности. Важной особенностью DLSS является интеграция шумоподавления и адаптивного сглаживания, что улучшает качество изображения, особенно в динамических сценах. Шумоподавление снижает артефакты, возникшие при апскейлинге, а адаптивное сглаживание помогает сохранить четкость на объектах с высокой детализацией [13].

Апробация

Функция **generate_image_from_text** (рисунок 1) определяет идентификатор модели и выбирает устройство (CPU или GPU), на котором будет выполняться генерация. Затем создается объект пайплайна, загружающего модель с заданным типом данных, и выполняется генерация изображения на основе текстового запроса. Полученное изображение сохраняется в файл.

```
def generate_image_from_text(prompt,
                             output_file='generated_image.png'):
    """ Генерирует изображение на основе текстового запроса
    с использованием модели Stable Diffusion.
    Аргументы:
    prompt (str): Текстовый запрос, на основе которого будет
    сгенерировано изображение.
    output_file (str): Имя файла для сохранения
    сгенерированного изображения.
    Возвращаемые значения:
    None: Изображение сохраняется на диск.
    """

    # Определение идентификатора модели
    model_id = "CompVis/stable-diffusion-v1-4"

    # Определение доступного устройства: CUDA, MPS (для
    ROCm) или CPU
    device = torch.device("cuda" if torch.cuda.is_available() else
    "mps" if torch.backends.mps.is_available() else "cpu")

    # Выбор типа данных: float16 для GPU, float32 для CPU
    dtype = torch.float16 если device.type != "cpu" else
    torch.float32
    # Создание пайплайна генерации изображения с
    заданным типом данных
    pipeline = StableDiffusionPipeline.from_pretrained(model_id,
    torch_dtype=dtype)
    pipeline.to(device)

    # Генерация изображения на основе текстового запроса
    if device.type == "cpu":
        image = pipeline(prompt).images[0]
    else:
        with torch.autocast(device.type):
            image = pipeline(prompt).images[0]

    # Сохранение изображения в файл
    image.save(output_file)
```

Рисунок 1 – Код использования функции
generate_image_from_text

Код автоматически определяет доступное устройство, будь то CPU, GPU с поддержкой CUDA или GPU с поддержкой ROCm, и использует его для выполнения модели. Это делает код более универсальным и удобным для работы на различных системах. На рисунке 2 приведен результат работы функции при запросе «Ландшафт с горами и рекой». Время генерации на CPU составило 5:48.



Рисунок 2 – Пример генерации рисунка по тексту

Заключение

В данной статье рассмотрены основные проблемы, с которыми сталкиваются исследователи и практики при генерации изображений с использованием нейронных сетей. Основными из них являются недостаточное качество результата на выходе, большое время для выполнения алгоритмов и обучения, а также высокая вычислительная сложность, требующая значительных ресурсов для обучения. Это создает вызовы не только с точки зрения вычислительной мощности, но и с точки зрения энергопотребления и экологической устойчивости.

Однако, несмотря на эти проблемы, исследования в области генерации изображений с использованием искусственного интеллекта продолжают активно развиваться. Перспективы дальнейшего развития включают в себя улучшение качества сгенерированных изображений, увеличение их разнообразия и реалистичности, а также сокращение требуемых ресурсов для обучения.

В будущем, исследователи могут сосредоточиться на разработке более эффективных алгоритмов и архитектур нейронных сетей, которые будут более энергоэффективными и менее требовательными к ресурсам. Также важным направлением может

стать разработка методов оптимизации, улучшающих процесс обучения. Кроме того, исследования в области использования специализированных аппаратных ускорителей и облачных вычислений могут привести к существенным улучшениям в этой области.

Направлением дальнейших исследований является разработка модифицированного алгоритма генерации сложной сцены, использующего для формирования фоновой поверхности средства ИИ и генерирующего быстроизменяющиеся изображения переднего плана традиционными методами.

Литература

1. Зори, С. А. Реалистичная визуализация трехмерных объектов и сцен с использованием технологий объемного отображения / С. А. Зори, Е. А. Башков // Известия ЮФУ. Технические науки. Тематический выпуск «Компьютерные и информационные технологии в науке, инженерии и управлении». - 2012. - № 5(130). - С. 133-137.
2. Зори, С. А. Использование средств аппаратной поддержки для повышения производительности систем 3D-пространственной визуализации / С. А. Зори, А. Я. Аноприенко, Р. В. Мальчева, О. А. Авксентьева // Информатика и кибернетика. - Донецк: ДонНТУ, 2019. - № 1 (15). - С. 5-12.
3. Хайкин С. Нейронные сети: полный курс / С. Хайкин – 2-е издание. Пер. с англ. – М.: Издательский дом «Вильямс», 2006. – 1104 с.: ил.
4. Goodfellow, Ian, et al. Generative adversarial nets // Advances in neural information processing systems, 2014.
5. Zhang, R. The unreasonable effectiveness of deep features as a perceptual metric / R. Zhang // Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2018. –
6. Постолиит, А. В. Основы искусственного интеллекта в примерах на Python. Самоучитель / А. В. Постолиит. – 2-е изд., перераб. и доп. – СПб.: БЧВ-Петербург, 2023 – 448 с.: ил.
7. Srivastava, Nitish, et al. Dropout: a simple way to prevent neural networks from overfitting // The Journal of Machine Learning Research 15.1, 2014. P. 1929-1958.
8. Shorten, Connor, and Taghi M. Khoshgoftaar. A survey on image data augmentation for deep learning // Journal of Big Data 6.1, 2019. – P. 60.
9. Howard, Andrew G., et al. Mobilenets: Efficient convolutional neural networks for mobile vision applications // arXiv preprint arXiv:1704.04861, 2017.
10. Wu, Yuhao, et al. Progressive Growing of GANs for Improved Quality, Stability, and Variation // arXiv preprint arXiv:1710.10196, 2017.

11. Howard, Andrew G., et al. Mobilenets: Efficient convolutional neural networks for mobile vision applications // arXiv preprint arXiv:1704.04861, 2017.

12. Google AI Blog. Google supercharges machine learning tasks with TPU custom chip [Electronic resource] - URL:

<https://ai.googleblog.com/2016/05/announcing-tpu-tensor-processing-unit.html>)

13. EIZO Rugged Solutions and NVIDIA Turing Push the Boundaries of Rugged AI [Electronic resource] - URL: https://www.unmannedsystemstechnology.com/wp-content/uploads/2021/01/NVIDIA_TuringGPU_whitepaper2021.pdf

Мулявин Д. Е., Мальчева Р. В., Койбаш А. А. Расширение возможностей систем генерации изображений путем использования нейронных сетей. Статья представляет обзор основных проблем, с которыми сталкиваются исследователи и практики при генерации изображений с использованием нейронных сетей. Рассматриваются ключевые аспекты, такие как вычислительная сложность и качество сгенерированных изображений. Предлагаются потенциальные решения для данных проблем, включая оптимизацию архитектур нейронных сетей, применение методов оптимизации и использование специализированных аппаратных ускорителей. Подводятся итоги перспектив развития исследований в данной области, а также указываются направления для будущих исследований и инноваций.

Ключевые слова: генерация изображений, нейронные сети, архитектура нейронных сетей, проблема качества изображений, задача оптимизации вычислений, ограничение вычислительной мощности.

Mulyavin D. E., Malcheva R. V., Koibash A. A. Expanding the capabilities of image generation systems by using neural networks. The article provides an overview of the main challenges faced by researchers and practitioners in generating images using neural networks. Key aspects such as computational complexity, power consumption and quality of the generated images are discussed. The paper also suggests potential solutions to these problems, including optimization of neural network architectures, application of optimization techniques, and use of specialized hardware gas pedals. The prospects for research in this area are summarized, and directions for future research and innovation are outlined.

Keywords: image generation, neural networks, neural network architecture, image quality problem, computational optimization problem, computational power limitation.

Статья поступила в редакцию 08.06.2024
Рекомендована к публикации профессором Зори С. А.