ВВЕДЕНИЕ

Данный курсовой проект направлен на углубленное изучение учебной дисциплины «Искусственный интеллект».

Целью данной работы поставлена задача создать демонстрационный набор стилизованных посредством переноса стилей фотографий с помощью нейросетей.

Демо – жанр компьютерного искусства, представляющий собой мультимедийную презентацию. Демо создаются в целях демонстрации возможностей и знаний в области программирования, компьютерной графики, 3D-моделирования или написания музыки.

Стилизация изображений и перенос стилей с помощью нейросетей (англ. neural style transfer) – техника обработки изображений, которая использует глубокое обучение и искусственные нейронные сети для изменения стиля (цветовая палитра, текстура, композиция и прочие художественные элементы) изображения, сохраняя при этом его содержание и структуру. Эта техника позволяет художникам, фотографам и дизайнерам легко преобразовывать обычные фотографии в произведения искусства, перенося стиль известных художников или создавая уникальные визуальные эффекты. Является старой проблемой, которая в прошлом решалась с трудом, но благодаря современным нейросетям, всего лишь правильно выставив необходимые параметры, возможно радикально преобразовать свою фотографию.

Задачи, которые ставятся в данной работе:

1. Провести анализ техники переноса стилей, разобраться с используемыми технологиями;
2. На основе проведенного анализа создать набор стилизованных изображений.

Актуальность данной технологии определяется спросом на обработку фотографий и даже видео. Результат в данном случае получается более

быстрым и, при нужных настройках, даже более аккуратным, нежели ручная обработка или отрисовка в специализированных программах.

В ходе данной курсовой работы необходимо выяснить, как именно работает рассматриваемая технология, какие этапы предстоит пройти для обработки изображения и изучить специфику работы используемой программы.

После аналитики требуется собрать целевые изображения и обработать их с помощью нейросетей, выполнив конечную цель по созданию демонстрационного набора.

Работа разделена на две части: аналитическая и проектная.

Все результаты работы, включая саму курсовую, будут доступны в публичном репозитории GitHub по ссылке «https://github.com/MMMdems/CourseWork-NeuralStyleTransfer-PI41».

Курсовая работа выполняется на основе уже полученных знаний и умений.

ГЛАВА 1
АНАЛИТИЧЕСКАЯ ЧАСТЬ

1.1. Stable Diffusion

Для стилизации изображений мною была выбрана нейронная сеть Stable Diffusion.

Stable Diffusion – программное обеспечение с открытым исходным кодом для создания изображений с помощью ИИ. Разработано группой компаний CompVis в Мюнхенском университете при содействии Runway, EleutherAI и LAION, и выпущено в 2022 году [15].

Генерация изображений при помощи ИИ – одна из самых новых возможностей искусственного интеллекта, поражающая своими результатами многих людей. Способность создания потрясающих изображений на основе текстовых описаний похожа на магию: компьютер становится всё ближе к идеалу и порой сложно отличить работу ИИ от человеческой. Выпуск Stable Diffusion стал важной вехой в этом развитии, поскольку высокопроизводительная модель оказалась доступной широкой публике (производительная с точки зрения качества изображения, скорости и относительно низких требований к ресурсам и памяти) и удобна в использовании [13].

Генеративная модель достаточна гибка и позволяет создавать изображения множеством способов [9]. Основных сценария использования у программного обеспечения всего два: создание изображений на основе текста (text2img) и на основе входного изображения и текста (img2img) [8]. Для переноса стиля можно использовать любой из них благодаря полезным расширениям [12].

Для краткого представления процесса необходимо разобраться во внутреннем устройстве модели. Stable Diffusion – сложная не монолитная система, состоящая из множества компонентов и моделей (рис. 1.1). Внутри находятся два основных компонента:

1. Компонент понимания текста, преобразующий текст в цифровой вид для передачи заложенного в него смысла. В его основе лежит специальная языковая модель Transformer (в Stable Diffusion используется модель ClipText на основе GPT), которая получает на вход текст, а на выходе выдаёт список чисел (вектор), описывающий каждое слово (токены) в тексте. Эта информация далее передаётся во второй компонент.

2. Генератор изображений, который производит обработку в два этапа. Первым делом он в определенное количество шагов (параметр steps) создаёт информацию об изображении посредством диффузии, действуя полностью в латентном пространстве изображения (это существенно ускоряет работу по сравнению с предыдущими моделями диффузии, работающими в пространстве пикселей). Данный этап выполняется с помощью нейросети UNet и алгоритма планирования. Последним этапом выступает декодер изображений, который на основе полученной информации создаёт готовое пиксельное изображение.

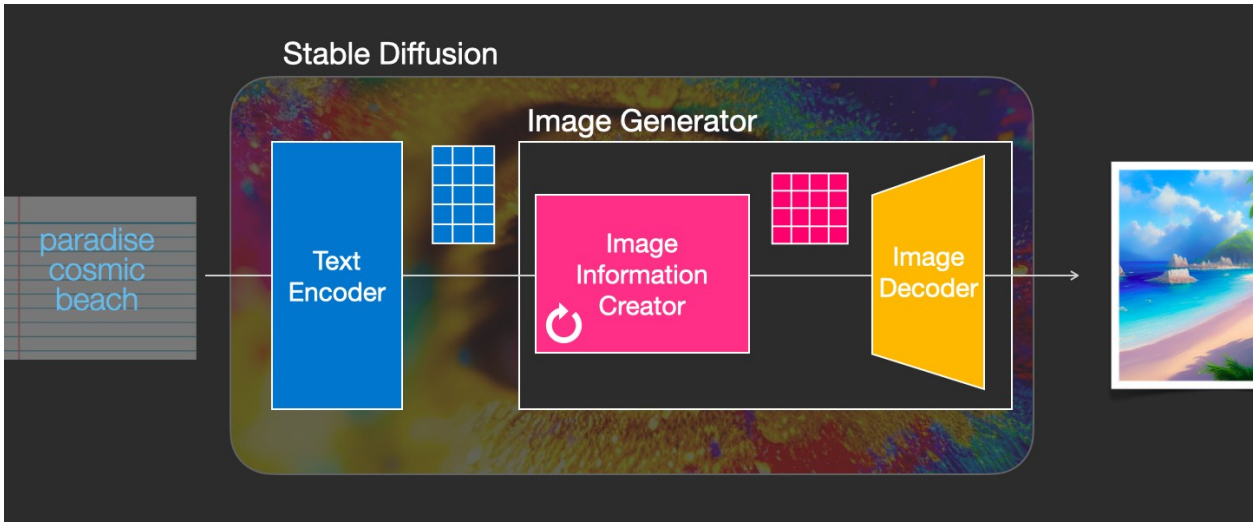


Рис 1.1. Визуальное представление генерации изображения

Диффузия – это процесс, выполняемый внутри этапа создания информации изображения. Имея токены, описывающие текст, и случайный начальный массив информации (шум), создаётся массив информации

изображения (latent) для декодера (рис. 1.2). Диффузия выполняется поэтапно, с каждым шагом добавляя всё больше релевантной информации из токенов, а каждый новый latent всё больше напоминает введенный текст [14].

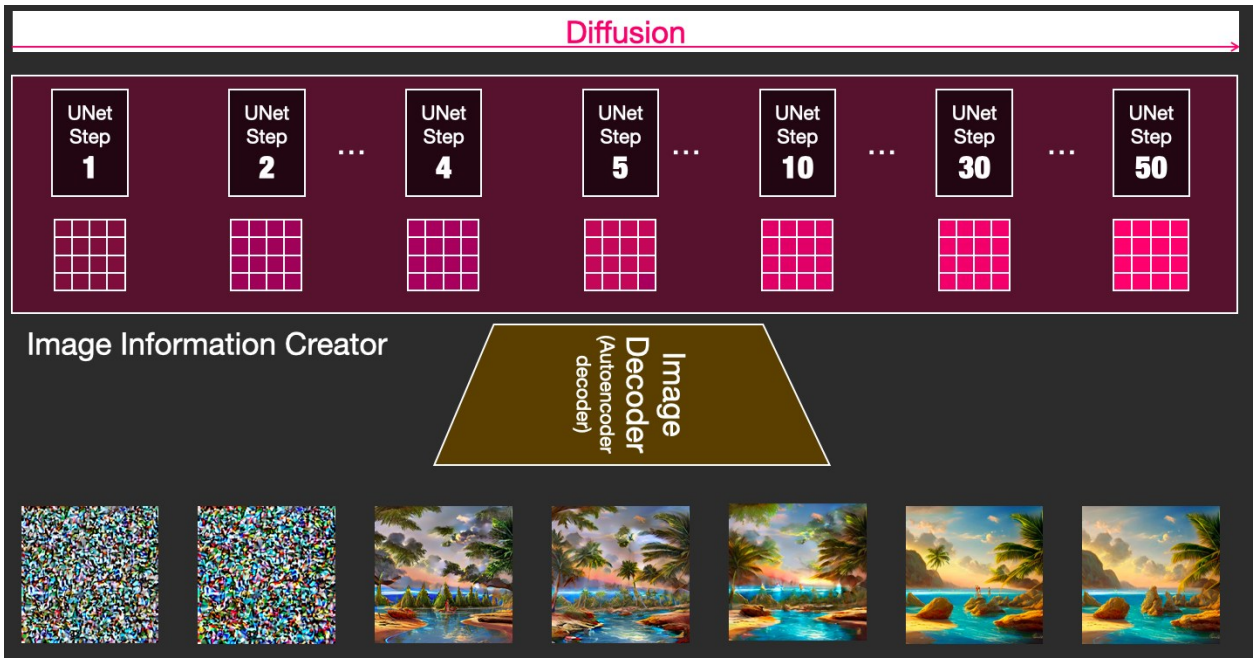


Рис. 1.2. Визуальное представление диффузии

Чтобы воспользоваться Stable Diffusion необходимо его установить и научиться управлять генерацией изображения. Вот пошаговая инструкция по установке:

1. Скачать со страницы GitHub самого популярного web-интерфейса Stable Diffusion (https://github.com/AUTOMATIC1111/stable-diffusion-webui/releases/tag/v1.0.0-pre) zip архив «sd.webui.zip».
2. Распаковать его в любое удобное место на компьютере (из условий только Windows 10+ и видеокарта от Nvidia).
3. Запустить в распакованной папке файл «update.bat» и дождаться выполнения обновления.
4. Запустить Stable Diffusion через файл «run.bat» и пользоваться в окне открывшегося браузера.

Из коробки в Stable Diffusion есть множество способов влиять на процесс генерации изображения. Вот основные из них:

1. Промпты (англ. prompts) и параметр CFG Scale (англ. Classifier Free Guidance Scale);
2. Метод семплирования (англ. Sampling Method) и количество его шагов (англ. Sampling Steps);
3. Модель (англ. Checkpoint);
4. LoRA (англ. Low-Rank Adaptation);
5. Текстовая инверсия или эмбеддинг (англ. Textual Inversion, Embedding).

Рассмотрим каждый из них по порядку.

Промпты – это текстовые запросы от пользователя к нейросети для получения ответа на основе этого запроса. Позитивные промпты используются для задания контекста или указания на то, что ожидается от конечного изображения в ответе. Негативные промпты используются для указания того, чтобы определенные элементы или характеристики изображения не были включены в конечное изображение. CFG Scale (направляющая шкала без классификатора) – параметр следования промптам, который определяет вариативность генерации и точность следования текстовому запросу. Важно помнить, что порядок слов в запросе имеет важное значение. Промпт следует писать, задавая себе уточняющие вопросы: Картина это или фото? Что находится на изображении? Какие детали стоит уточнить? Какое освещение должно быть? Какое окружение на изображении? Какие цвета стоит использовать? В каком конкретном художественном стиле будет картинка? Также можно написать некоторые модификаторы изображения, такие как стиль фотографии (как было снято фото, вид обработки, камера, линза, освещение и т.д.), средство рисунка (масляная картина, акварель, граффити, дерево, ткань и т.д.), имя художника (портретист, пейзажист, аниме, концепт-арт, фотограф и т.д.), стиль иллюстрации (оригами, 3D рендер,

вектор, детская книжка, комикс, постер и т.д.), эмоции (энергичность, романтика, спокойствие, печаль, меланхолия и т.д.), эстетику (кислотность, сон, брутализм, пост апокалипсис, барокко, нуар и т.д.) и некоторые «магические» слова (HDR, UHD, 64K, highly detailed, studio lighting, professional и др.) (рис. 1.3) [10].

A painting of a cute goldendoodle in the sky, wearing a suit, natural light, with bright colors, by Studio Ghibli

Рис. 1.3. Пример составления промпта

Метод семплирования – алгоритм, по которому нейронная сеть убирает шум с изображения. Зачастую выбор метода — это компромисс между скоростью и точностью. На каждом шаге задача семплера заключается в создании изображения с уровнем шума, соответствующим графику шума. Увеличение количества шагов приводит к уменьшению снижения шума на каждом шаге и, соответственно, уменьшает вероятность ошибки при усечении. Приписки «SDE» и «a» в названии означают, что результат при повторении генерации, как правило, не будет сходиться с предыдущим. Если выбор пал на что-то быстрое, универсальное и с нормальным качеством, то выделяются в этом плане семплеры «DPM++ 2M Karras» и «UniPC» с 20-30 шагами. Для изображений хорошего качества рекомендуются «DPM++ SDE Karras» и «DDIM» с 10-15 шагами. Если нужно что-то простое и быстрое, то подойдут семплеры «Euler» и «Heun» с 5-10 шагами (рис. 1.4).

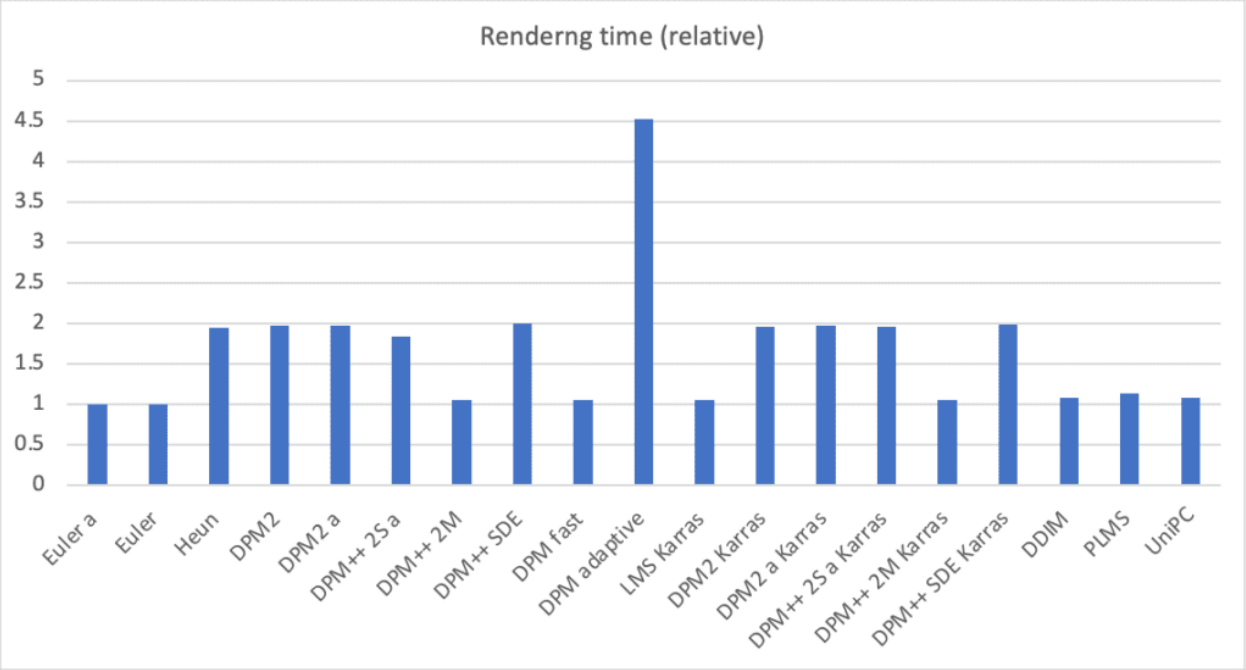


Рис. 1.4. Сравнение относительной скорости работы семплеров

Модели Stable Diffusion или же файлы контрольных точек – это предварительно настроенные веса базовых моделей. Они предназначены для создания либо общих изображений, либо изображений определенного стиля. Тип изображений, которые может генерировать модель, зависит от данных, используемых во время ее обучения. Например, модель, обученная исключительно изображениям кошек, будет хорошо генерировать только изображения кошек. Этот принцип становится совершенно очевидным, если мы рассматриваем создание изображений в стиле аниме. Для примера возьмем две модели, которым было поручено создать образ девушки из аниме (рис. 1.5). Первая модель, недостаточно натренированная на данных по аниме, изо всех сил пытается создать убедительный образ, но получает только искаженное изображение. И наоборот, вторая модель, прошедшая обучение по аниме, успешно создает образ девушки из аниме. Сравнение этих результатов подчеркивает важность обучающих данных для формирования возможностей моделей.

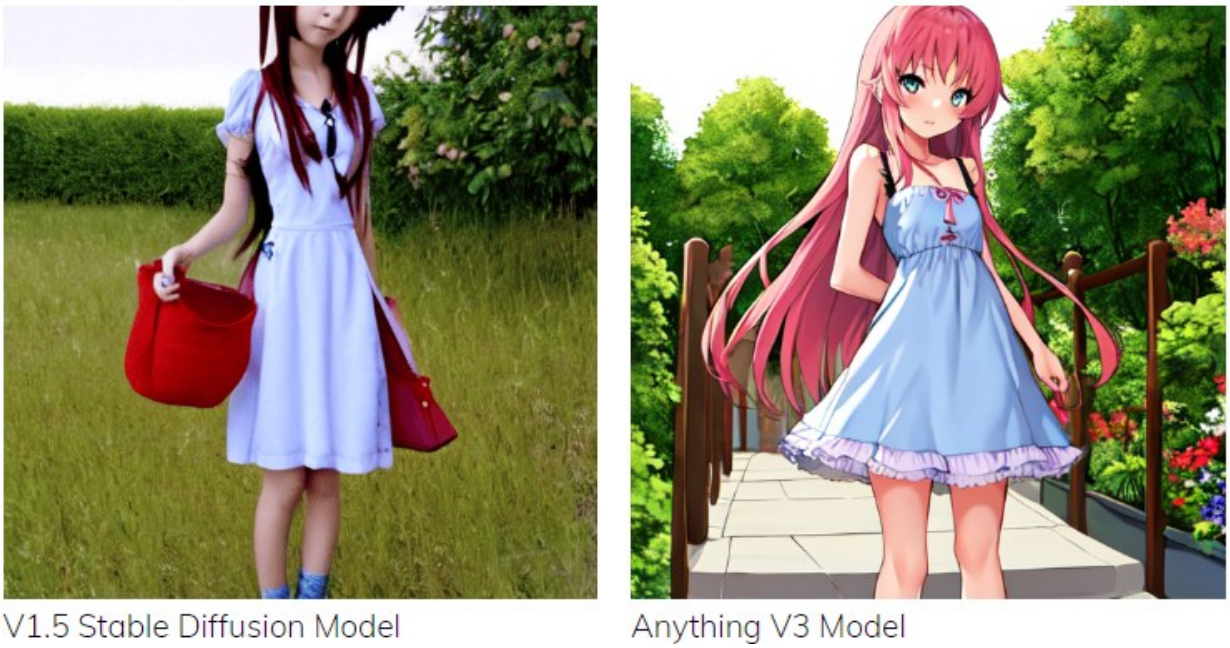


Рис. 1.5. Сравнение разных моделей Stable Diffusion

LoRA или низко ранговая адаптация – это небольшое дополнение, которое ставится поверх моделей и помогает создавать изображения в определенном стиле, определенных персонажей или определенные предметы. Изначально стандартная модель обучается на определенном большом наборе изображений и не может, например, знать, как выглядит Карлсон (или любой другой персонаж), но, с помощью предварительно обученной на изображениях Карлсона LoRA, возможно научить нейросеть выдавать по ключевому промпту нужного нам персонажа. Ее основное преимущество заключается в том, что позволяет обучить нейросеть новой концепции, не требуя много времени и сотни изображений.

Текстовая инверсия или по-другому эмбеддинг – это некий аналог очень точного промпта, который позволяет одним словом выразить множество других. Если обычный промпт просто выдаёт картинку, то эмбеддинг выдаёт нейросети картинки, которые мы хотим получить и просит модель сказать, какой промпт должен быть. Например, мы собрали большой набор картинок, на которых явно плохо нарисованы руки, акцентировали на этом внимание при

тренировке эмбеддинга и получили файл. Проще говоря этот файл является точно таким же словом, как и другие слова в текстовом запросе. Именно поэтому они так мало весят - потому что это набор нужных слов, только напрямую в цифрах. И вместо того, чтобы писать в негативном промпте bad hands, extra fingers и т.д. – мы пишем наш эмбеддинг, который нейросеть понимает как ещё одно слово, означающее "руки очень плохо нарисованы, не надо так рисовать".

Помимо встроенных способов влиять на генерацию хочу добавить в этот список расширение «stable-diffusion-aesthetic-gradients» (установку см. в приложении А), которое позволяет использовать эстетические эмбеддинги. По своей сути это те же самые эмбеддинги, но встраиваемые по другому алгоритму (рис. 1.6) [6].

Рис. 1.6. Справа с эстетичным эмбеддингом, слева - без

1.2. ControlNet

Изначально в Stable Diffusion создание изображений всегда начинается с «чистого холста», а передача во входные параметры нашей фотографии (img2img) не позволяет добиться ни качественного переноса стиля, ни удержания даже половины деталей оригинала (многое алгоритм додумывает сам) [1]. Для стилизации изображения нам необходимо максимально сохранить содержание и композицию исходного изображения. С этой задачей призвано справиться расширение «sd-webui-controlnet» (установку см. в приложении А) [7].

ControlNet — это архитектура нейронной сети, которая позволяет управлять процессом генерации изображения в диффузионных моделях, добавляя определённые входные условия, такие как форма, глубина, поза объекта и некоторые другие. Эта сеть управления требует всего 23% дополнительной памяти и 34% дополнительного времени по сравнению с исходной моделью взамен на потрясающие способности по контролю результата.

Перевод изображения в изображение — это метод, который изучает сопоставление между изображениями в разных областях, таких как художественные работы, фотографии или эскизы. Для достижения этой цели использовались различные типы алгоритмов, такие как условно порождающие нейронные сети, методы авторегрессии и многомодельные методы. Новейшие методы преобразования можно найти в Taming Transformer, Palette и PITI. Они используют такие методы, как VIT (Vision Transformers), файнтюнинг и основаны на оптимизации управления процессом диффузии для получения результатов. ControlNet отличается от этих методов тем, что нацелена на управление диффузионными моделями с учётом конкретных входных условий, а не на изучение сопоставления между изображениями.

Для генерации с ControlNet всегда нужна картинка-референс — она выступает как трафарет. Расширение генерирует на основе оригинала

наброски разных форматов, а потом использует их как основу для дальнейшей генерации. Это решает сразу несколько известных проблем нейросетей с генерацией групп людей, текста, рук, глаз и других мелких деталей. Раньше Stable Diffusion генерировал пальцы на основе «шума» из миллионов картинок разных пальцев и это зачастую давало странные результаты - теперь же она имеет четкий трафарет и может повторить контур руки [16].

В ControlNet входят множество разных моделей. Перед генерацией нужно подобрать ту, которая лучше всего подходит для задачи, а затем загрузить в нее изображение. Также в настройках расширения можно включить мульти контроль, что позволит использовать сразу несколько моделей при генерации.

Вот краткий разбор каждой модели:

Canny — рисует тонкой линией набросок изображения (рис. 1.7). Так она довольно четко выделяет очертания объекта, сохраняя многие мелкие элементы и позволяет заимствовать у картинки композицию и стилистику, но менять детали — скажем, лицо персонажа или цвета на картинке. У модели есть параметры «Canny Low Threshold» и «Canny High Threshold», которые отвечают за степень захвата линий.

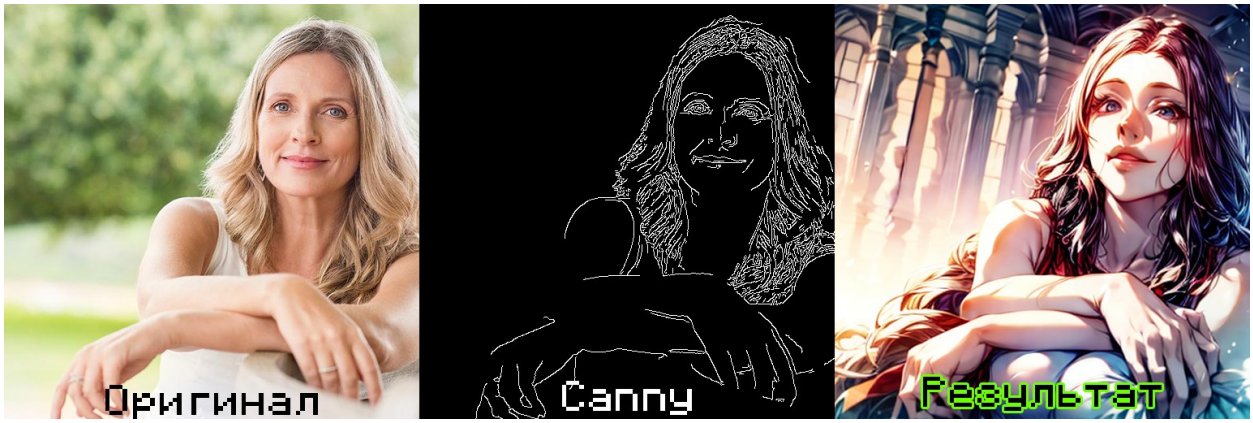


Рис. 1.7. Пример работы модели Canny

Depth — рисует карту глубины, которая показывает, какие объекты находятся ближе (белый цвет), а какие дальше (чёрный цвет) (рис. 1.8).

Полезно, когда нужно сохранить композиционную глубину и показать, как должны отображаться объекты в перспективе. Всего на выбор есть 4 алгоритма вычисления глубины: leres – пытается захватить побольше объектов; leres++ - захватывает еще больше подробностей; midas – выделяет только основной план, немного игнорируя задний; zoe – модифицированный midas, захватывающий больше деталей.

Рис. 1.8. Пример работы модели Depth (leres++)

NormalMap — рисует карту нормалей изображения, определяя положение объекта в трехмерном пространстве и как на него падает свет (рис. 1.9). Отлично подходит для 3D (например, можно легко сменить материал на золото, ткань, дерево и прочее) или реалистичных изображений. Модель предлагает два алгоритма составления карты нормалей – normalbae для захвата всей сцены или midas для захвата переднего плана.

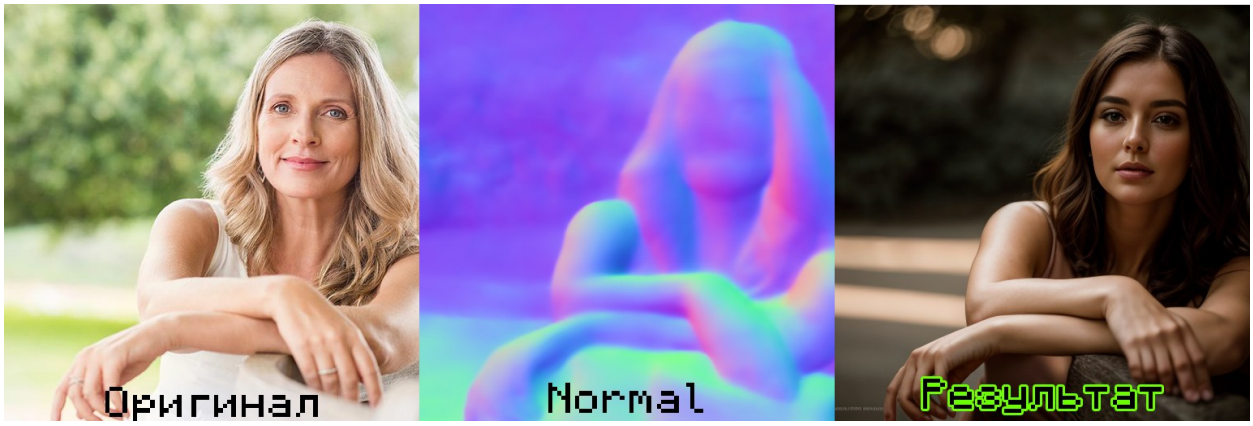


Рис. 1.9. Пример работы модели NormapMap (normalbae)

OpenPose — создает «скелет», довольно точную позу объекта (лучше всего работает на людях) (рис. 1.10). У модели есть редактор «костей», где можно перетаскивать части тела и делать свои позы. В настройках можно выбрать, какая часть тела нас интересует – лицо, всё тело, руки и ещё парочка вариаций.

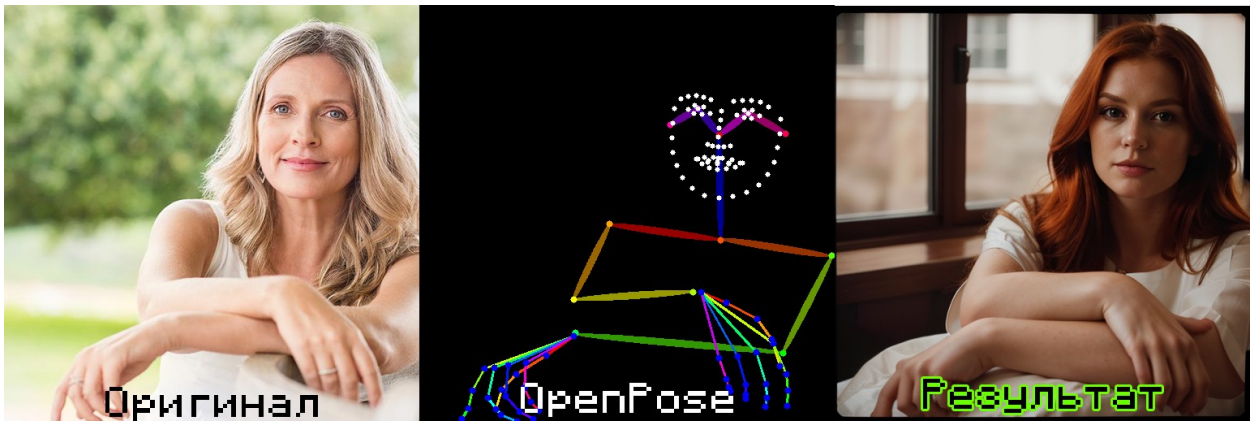


Рис. 1.10. Пример работы модели OpenPose (dw_openpose_full)

MLSD — обнаруживает прямые линии на изображении (рис. 1.11). Полезен для генерации картинок на основе референсов с прямыми линиями: дизайна интерьеров, зданий, уличных сцен. Модель предлагает настройки «Value Threshold» и «Distance Threshold», чтобы подправить значения алгоритма поиска прямых линий под своё изображение.

Рис. 1.11. Пример работы модели MLSD

LineArt — создает детализированные или грубые наброски изображения (рис. 1.12). Отлично подходит для реализма или скетчей. На выбор предоставляется несколько алгоритмов в зависимости от того, насколько детализированным должен быть набросок.

Рис. 1.12. Пример работы модели LineArt (realistic)

SoftEdge — размыто и неаккуратно очерчивает объекты на фото так, чтобы результат казался работой человека (рис. 1.13). Такой набросок хорошо подходит для картин или изменения стилистики. В настройках можно выбрать алгоритмы hed или pidi (менее четкий), по-разному размывая границы объектов.

Рис. 1.13. Пример работы модели SoftEdge (hed)

Sketch — создаёт что-то наподобие векторного рисунка, полученного в результате перевода пиксельного изображения в «.svg» формат в программе

Adobe Illustrator (рис. 1.14). Достаточно плохо сохраняет детали, но в некотором роде «развязывает» руки нейросети, позволяя кардинально менять результат на каждой итерации. Алгоритмов создания подобных каракуль четыре и каждый отличается уровнем детализации.

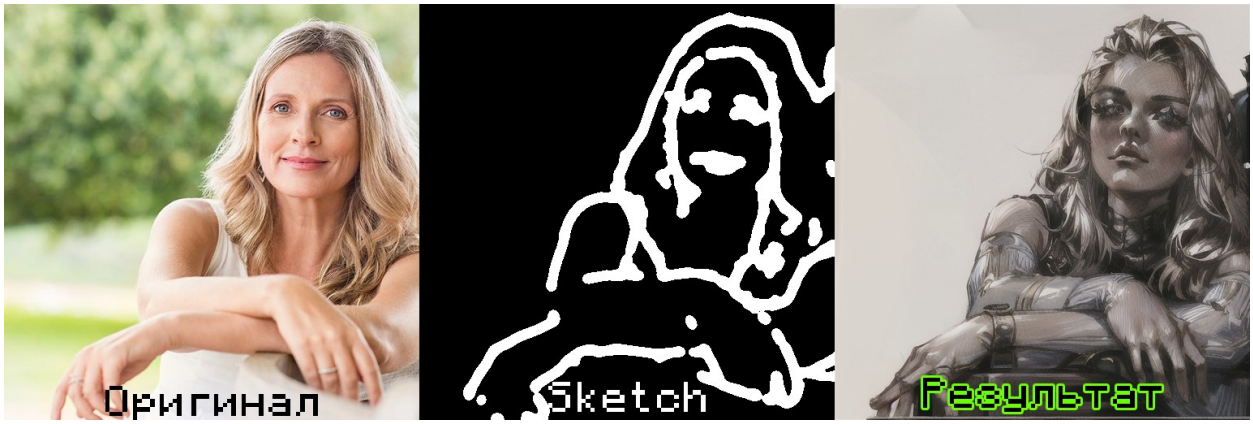


Рис. 1.14. Пример работы модели Sketch (scribble_hed)

Segmentation — разделяет рисунок на некоторые области на основе его содержания, классифицируя разными цветами объекты или регионы (рис. 1.15). Хорошо подходит для того, чтобы сохранить исходную композицию на фото, но при этом создать что-то совершенно не похожее на оригинал. Провести семантическую сегментацию можно тремя алгоритмами: ofade20k, ufade20k, ofcoco (самый детальный).

Рис. 1.15. Пример работы модели Segmentation (ofcoco)

Shuffle — довольно случайным образом переносит цветовую гамму с исходного изображения на результат генерации (рис. 1.16). «Случайность» сгенерированной цветовой гаммы зависит от параметра seed.

Рис. 1.16. Пример работы модели Shuffle

Tile — используется для увеличения разрешения изображения, но его работа в данной курсовой нас не интересует.

Inpaint — представляет собой улучшенный вариант встроенного inpaint Stable Diffusion. Его работа нас также не интересует, хотя и позволяет получать различные вариации нашего исходника.

InstructPix2Pix — позволяет при помощи текстового запроса указать, как нужно изменить изображение (рис. 1.17). Такая обработка позволяет сохранить детали (неконтролируемо), но при этом поменять цвета или добавить новые, например изменить освещение, погоду и прочие эффекты.

Рис. 1.17. Пример работы модели IP2P

Reference — использует исходник в качестве источника информации для генерации изображения (рис. 1.18). Всего доступно три режима работы: reference_only (достает из исходника информацию наподобие промпта), reference_adain (пробует перенести стиль с исходника с помощью адаптивного алгоритма) и reference_adain+attn (комбинация двух предыдущих).

Рис. 1.18. Пример работы модели Reference (adain+attn)

Recolor — раскрашивает черно-белые изображения или перекрашивает цветные, предварительно обесцвечивая их (рис. 1.19). В выборе цветов зависит от промпта и выбранной модели.

Рис. 1.19. Пример работы модели Recolor

Revision — одна из новейших моделей, которая берет за основу концепт исходника и создаёт своё собственное изображение (рис. 1.20). Доступны всего две настройки: игнорировать ли промпты и добавлять ли шум.

Рис. 1.20. Пример работы модели Revision

T2I-Adapter и IP-Adapter — сторонние модели, интегрированные в ControlNet, которые позволяют переносить стиль с одного изображения на другое посредством встроенных нейросетей (рис. 1.21). T2I-Adapter также обладает дополнительным препроцессором, который создает цветовую палитру, используемую в исходнике (рис. 1.22) [17].

Рис. 1.21. Пример работы модели T2I-Adapter (style)

Рис. 1.22. Пример работы модели T2I-Adapter (color)

1.3. Альтернативные способы переноса стиля

Stable Diffusion является современнейшим и наиболее контролируемым инструментом для стилизации изображений, однако в этой области есть и более старые решения.

Первой и более старой, но всё же интересной программой я рассмотрю «neural-style-pt», которая расположена в виде репозитория GitHub по URL ссылке «https://github.com/ProGamerGov/neural-style-pt».

Neural-style-pt — это PyTorch реализация научных статей «Image Style Transfer Using Convolutional Neural Networks» (рус. - Передача стиля изображения с использованием сверточных нейронных сетей) и «A Neural Algorithm of Artistic Style» (рус. - Нейронный алгоритм художественного стиля) Леона Гэтиса, Александра Эккера и Матиаса Бетге. Суть представленного алгоритма состоит в объединении содержания одного изображения со стилем другого. PT в названии означает, что в алгоритм добавлена система передачи цвета Пити, что на самом деле не сильно меняет результаты работы оригинального алгоритма [3].

Наглядная демонстрация разницы в результатах (рис. 1.23 и рис. 1.24):

Рис. 1.23. Пример стилизации фото в Neural Style PT

Рис. 1.24. Пример стилизации фото в Stable Diffusion

Очевидно, что нейросети работают по разным алгоритмам, однако всё же возможно выделить ключевые отличия. Первая точно перенесла цвета и сохранила больше деталей от оригинала, однако заметно, что на изображении превалирует стиль неба, а не всей картины в целом. Stable Diffusion в этом примере хоть и ошиблась с цветами и добавила много деталей, однако сохранила общую композицию и содержание. В данном случае можно сказать, что Stable Diffusion является более продвинутой нейросетью, так как может повторить результат первой – необходимо только более точно выставить параметры, что может потребовать много времени для поиска оптимальных. В данном случае можно сказать, что Stable Diffusion является более сложной нейросетью – для достижения результата, подобного первой сети, ей может потребоваться более точная настройка параметров, что может занять дополнительное время для поиска оптимальных [5].

Вторая нейросеть, которую стоит рассмотреть — Deep Photo Style Transfer. Её исходный код находится в репозитории GitHub по URL ссылке «https://github.com/luanfujun/deep-photo-styletransfer» [2].

Deep Photo Style Transfer, как и предыдущая нейросеть, основана на работах Гэтиса (с большими переработками) и реализована на PyTorch. Основным моментом, на котором фокусируется данная нейросеть — строгая фотореалистичность, которая достигается за счет работы только в цветовом пространстве [4].

Сравнение результатов Deep Photo Style Transfer и Stable Diffusion:

Рис. 1.25. Пример стилизации фото в Deep Photo Style Transfer

Рис. 1.26. Пример стилизации фото в Stable Diffusion

Как видно, Deep Photo Style Transfer справилась с задачей лучше, добавив больше текстуры огня в конечное изображение. Stable Diffusion же сохранило флакон с духами в более нетронутом состоянии, только окрасив содержимое во что-то, похожее на отражения от огня [11].

Приведя в пример эти две нейросети, можно убедиться, что более старые решения не уступают в своих узкоспециализированных задачах более новой и комплексной Stable Diffusion, если даже не превосходят. Наконец можно сказать, что у всех трёх есть свои преимущества и недостатки, а выбор зависит больше от задач и целей, которые ставит перед собой пользователь данных программных решений.

1.4. Процесс переноса стиля в Stable Diffusion

Теперь, изучив инструментарий, который мы можем использовать для стилизации фото, можно приступить к составлению правил, по которым будут составляться изображения из демонстрационного набора (для повторяемости результата).

При генерации будут использоваться некоторые параметры, которые будут неизменными (стандартными) для всех изображений:

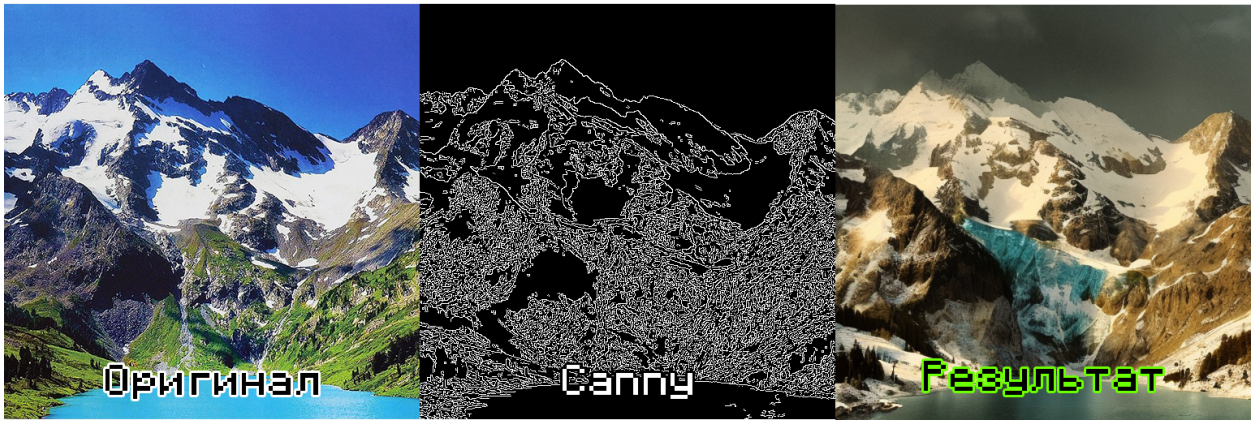
1. Размер изображения: 512 x 512 пикселей;
2. Все общие параметры ControlNet, не касающиеся самой модели.

В работе будут использованы модели, LoRA и эмбеддинги, скачанные с интернета. Инструкция по их установке и ссылки на скачивание будут приведены в соответствующих приложениях (см. Приложение Б и Приложение В).

Для расширения «Aesthetics Gradients» все эмбеддинги входят в базовый набор, поэтому ссылок на них не будет.

Формат (шаблон) выведения параметров, используемых при генерации изображения (рис. 1.27):

0. Изображения генерации (с подписями): исходное, стиль (если есть) и результат;
1. Позитивный промпт и негативный промпт;
2. Семплер, количество шагов, CFG Scale, seed;
3. Используемая модель Stable Diffusion;
4. Модули ControlNet, их параметры и изображения (с подписями), которые они создают, если таковые присутствуют;
5. (Опционально) Параметры расширения «Aesthetics Gradients».



Позитивный промпт: mountain landscape, scenery
Негативный промпт:
Семплер: DPM++ SDE Karras Количество шагов: 22 CFG Scale: 7 Seed: 2006212635
Модель: photon_v1
Модуль ControlNet 1: препроцессор Canny, модель control_v11p_sd15_canny,
Low-High Threshold: 100-200

Рис. 1.27. Пример демонстрационного изображения

ГЛАВА 2

ПРОЕКТНАЯ ЧАСТЬ

Для создания демонстрационного набора необходимо сначала определиться с изображениями, которые будут обработаны, и со стилями, в которые они будут преобразованы.

Начну со стилей, которые я хочу попробовать воплотить:

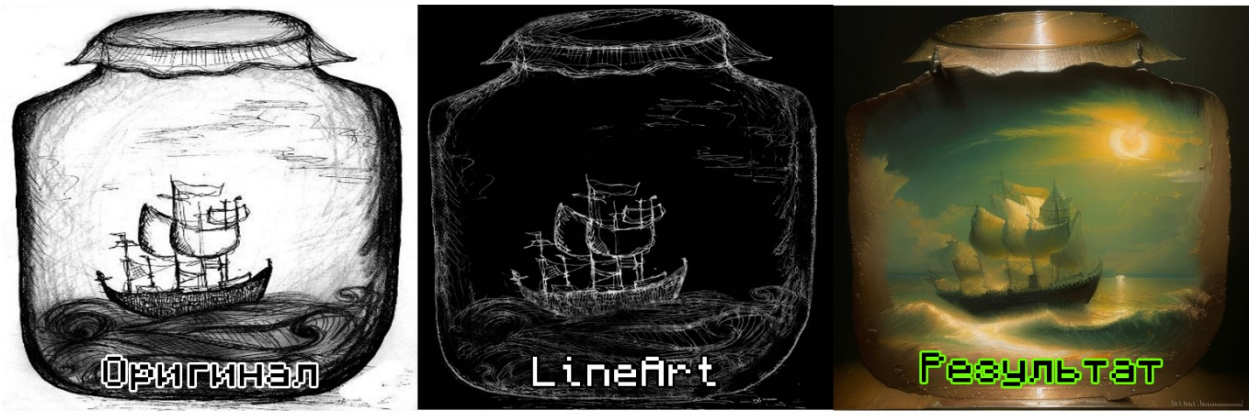
1. Картина знаменитого художника И.К. Айвазовского;
2. Черно-белая рисовка манги;
3. Стиль графики из игры Disco Elysium;
4. Картина в стиле китайской живописи;
5. Рисовка в духе старых постеров для фильмов ужасов;
6. Фотореализм;
7. Цифровой арт;
8. Рисовка в стиле мультфильмов от студии Ghibli;
9. Поп-арт комикс;
10. Dark art (Никола Самори).

Определившись со списком стилей, можно приступить к поэтапному созданию нашей небольшой коллекции, а именно: сначала собрать небольшую информацию о стиле, чтобы было понятнее на что должен быть похож результат, а затем уже создать непосредственно стилизованный рисунок.

Айвазовский (рис. 2.1).

Рис. 2.1. Пример работ И. К. Айвазовского

В первую очередь этот художник известен благодаря своим изумительным морским пейзажам, что значит преобладание на его картинах изображение неповторимой водной стихии. Стиль рисования Айвазовского можно охарактеризовать как близкий к реализму или даже гиперреализму, поэтому для более интересного результата и контраста необходимо подобрать некий достаточно простой рисунок – для этих целей я мой выбор пал на набросок карандашом, что в итоге дало действительно отличный результат (рис. 2.2).



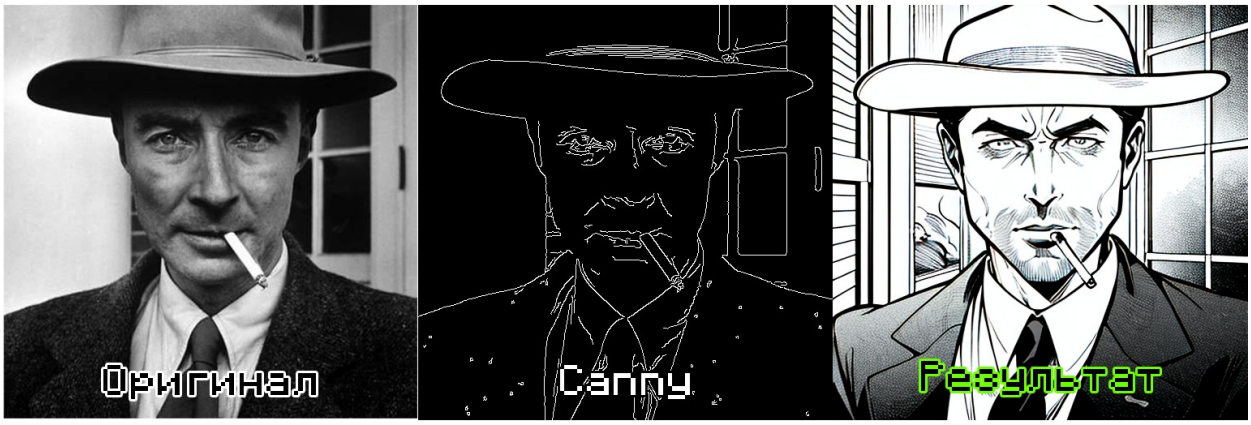
Позитивный промпт: painting by Aivazovsky
Негативный промпт:
Семплер: DPM++ SDE Karras Количество шагов: 20 CFG Scale: 7 Seed: 252965670
Модель: revAnimated_v122EOL
Модуль ControlNet 1: препроцессор Lineart-Anime, модель control_v11p_sd15_lineart
Aesthetic Gradients: aivazovsky 5 steps

Рис. 2.2 Демонстрационный набор 1 – Айвазовский

Манга (рис. 2.3).

Рис. 2.3. Пример рисовки манги

Манга является самым популярным видом комиксов со своеобразной черно-белой мультяшной рисовкой. У данного стиля можно выделить основные особенности: акцент на линиях, большие глаза, выразительность. Для обработки я подобрал фото Роберта Оппенгеймера (рис. 2.4).



Позитивный промпт: man detective with a cigarette, manga, lineart, monochrome, grayscale, <lora:animeoutlineV4_16:1>, Cowboy Bebop style
Негативный промпт: easynegative, lr, bad-artist, bad-artist-anime
Семплер: DPM++ SDE Karras Количество шагов: 22 CFG Scale: 7 Seed: 2084757152
Модель: AnythingV5Ink
Модуль ControlNet 1: препроцессор Canny, модель control_v11p_sd15_canny

Рис. 2.4. Демонстрационный набор 2 – Манга

Disco Elysium (рис. 2.5).

Рис. 2.5. Пример артов из Disco Elysium

Стиль в Disco Elysium можно охарактеризовать как масляные картины с контрастными мазками. Для стилизации был подобран фотопортрет (рис. 2.6).

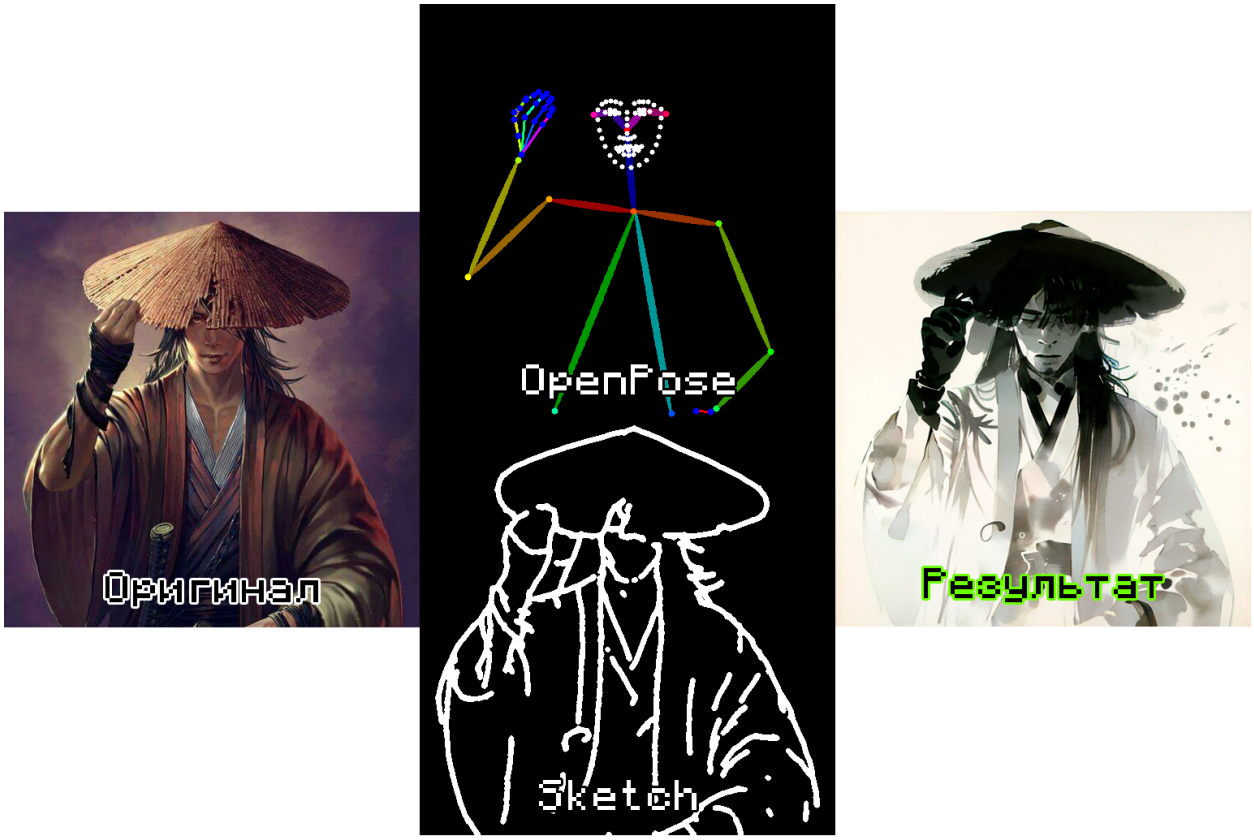
Позитивный промпт: masterpiece, best quality, 8k, oil painting, soft lighting, morning, ZaUm, elysiumChar, 1girl, ((expressionless)), portrait, face focus, <lora:DiscoElysiumStyle:1>, beautiful blue eye, <lora:detailed_eye_V2:0.2>
Негативный промпт: easynegative, badhandv4, watermark, sample watermark, logo, artist logo, patreon logo, lr, (low quality, worst quality:1.4), (monochrome, dof:1.2), bad anatomy, extra digits, long body, jpeg artifacts, signature, watermark, username, censor, text
Семплер: DDIM Количество шагов: 22 CFG Scale: 7 Seed: 2703468552
Модель: AnythingV5Ink
Модуль ControlNet 1: препроцессор SoftEdge, модель control_v11p_sd15_softedge
Модуль ControlNet 2: препроцессор NormalMap, модель control_v11p_sd15_normalbae

Рис. 2.6. Демонстрационный набор 3 – Disco Elysium

Китайская живопись (рис. 2.7).

Рис. 2.7. Пример китайской живописи

За основу данной стилизации я хочу взять работы тушью китайских художников 20 века. Оригиналом послужит рисунок самурая (рис. 2.8).



Позитивный промпт: <lora:zyd232_InkStyle_v1_0:1.3>, zydink, monochrome, grayscale, black ink sketch, 1boy, asian (young attractive man), non transparent tilted forward straw hat, looking at viewer, long hair, floating hair, hanfu, chinese clothes, long sleeves, (abstract ink splash:1.3), white background

Негативный промпт: watermark, bad anatomy, bad hands, bad-artist, bad-artist-anime, bad_prompt_version2

Семплер: DPM++ 2M Karras Количество шагов: 32 CFG Scale: 7 Seed: 297808180

Модель: KanpiroMixV20

Модуль ControlNet 1: препроцессор OpenPose, модель control_v11p_sd15_openpose

Модуль ControlNet 2: препроцессор Sketch, модель control_v11p_sd15_scribble, 1024px

Рис. 2.8. Демонстрационный набор 4 – Китайская живопись

Постер фильма ужасов (рис. 2.9).

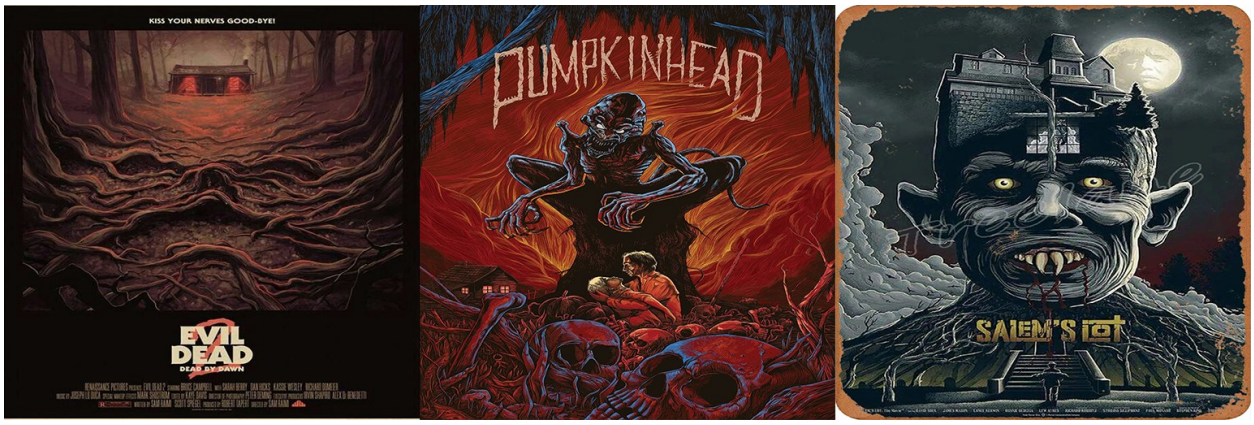


Рис. 2.9. Пример постеров фильмов ужасов

Отличительной чертой подобных постеров является упор на реализм, небольшая зернистость как в старых фильмах, масляные краски, гротескные силуэты и, конечно же, мрачность содержания. Исходником послужило фото женщины, закрывшей лицо руками (рис. 2.10)

Позитивный промпт: 1970`s horror book cover paper art style drawing of monster, by Zdzislaw Beksinski, methurlant, <lora:by_zdzislaw_beksinski_finetuned:0.4>, <lora:methurlant:0.4>, (intricate, atmospheric, surreal, gritty, cinematic, stylized, contrast, comic, eerie, stylized, cybernetic, futuristic, dystopian), film grain
Негативный промпт: (3D, Adorable, Anime, CGI, Cartoon, Cartoonish, Childish, Comic, Computer Generated, Cute, Happy, Joyful, Manga, Rendered), badhandv4
Семплер: Euler Количество шагов: 35 CFG Scale: 7 Seed: 2543306017
Модель: EpicRealism
Модуль ControlNet 1: препроцессор Depth_Zoe, модель control_v11p_sd15_depth

Рис. 2.10. Демонстрационный набор 5 – Постер фильма ужасов

Фотореализм (рис. 2.11).

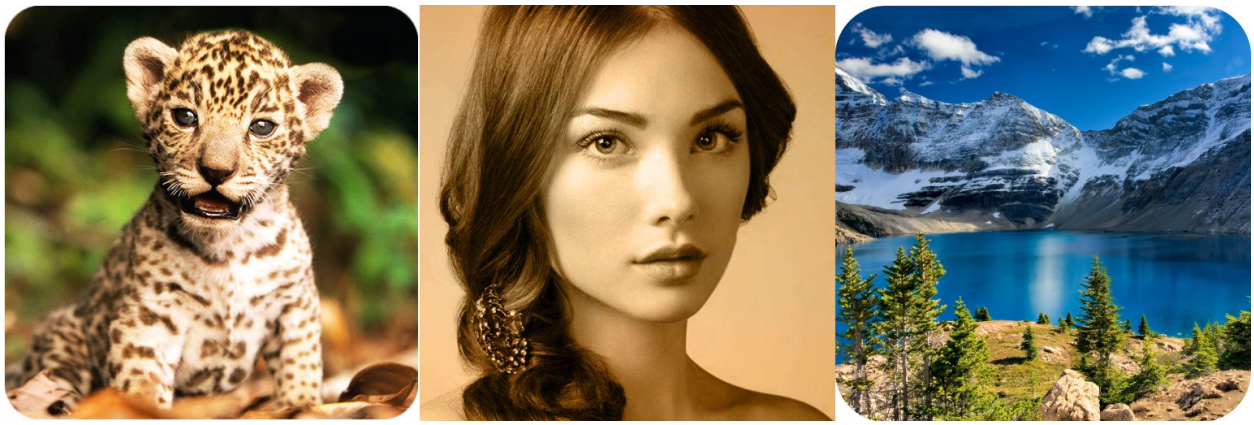


Рис. 2.11. Примеры фотореализма

Основой для стиля служат реальные фотографии. Стилизация выполнена на основе котенка в стиле Disney (рис. 2.12).

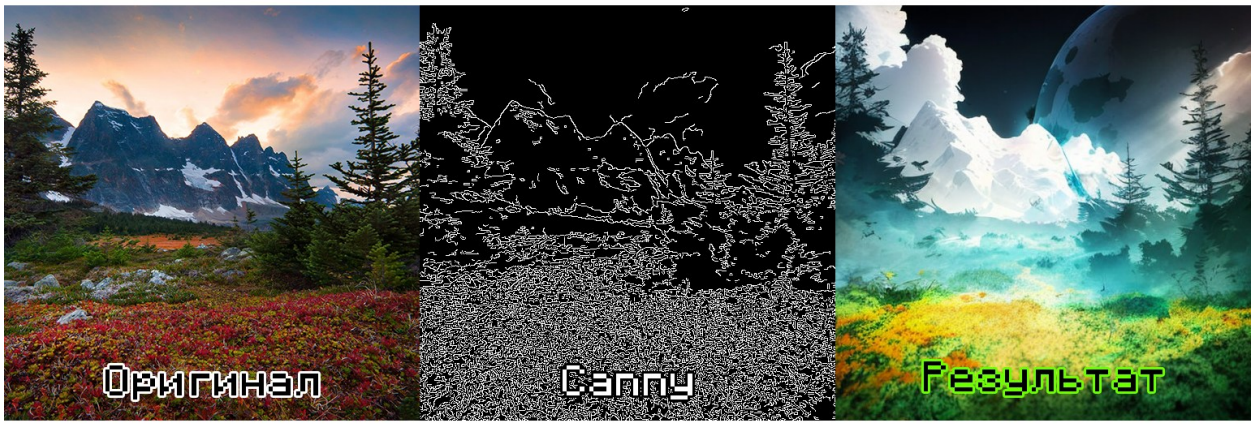
Позитивный промпт: black cat with (white belly, white paws, white tail`s tip), professional wildlife phography, cinematic, photorealistic, rim lighting, two tone lighting, dimly lit, 8k, ultra hd, extra high quality, HDR, UHD, 64K, studio lighting
Негативный промпт: bad_prompt_version2, ng_deepnegative_v1_75t, verybadimagenegative_v1.3, BadDream, UnrealisticDream, badhandv4, FastNegativeV2
Семплер: DPM++ SDE Karras Количество шагов: 50 CFG Scale: 6 Seed: 3528174820
Модель: KanpiroMix_V20
Модуль ControlNet 1: препроцессор Depth_Midas, модель control_v11p_sd15_depth
Модуль ControlNet 2: препроцессор Revision

Рис. 2.12. Демонстрационный набор 6 – Фотореализм

Цифровой арт (рис. 2.13).

Рис. 2.13. Примеры цифрового арта

В широком смысле цифровой арт это просто работы, созданные с использованием компьютерных технологий. Тем не менее про них можно сказать, что такие изображения являются более эффектными, нежели фотографии, из-за чего их часто ставят как обои рабочего стола. За основу стилизации взят горный пейзаж (рис. 2.14).



Позитивный промпт: scenery, landscape, 2d illustration, digital art, trend on artstation, lr, advntr
Негативный промпт: FastNegativeV2, ng_deepnegative_v1_75t, verybadimagenegative_v1.3, bad-artist, bad-artist-anime
Семплер: DPM++ SDE Karras Количество шагов: 20 CFG Scale: 7 Seed: 2300837838
Модель: Dreamshaper8
Модуль ControlNet 1: препроцессор Canny, модель control_v11p_sd15_canny

Рис. 2.14. Демонстрационный набор 7 – Цифровой арт

Ghibli (рис. 2.15).

Рис. 2.15. Пример работ студии Ghibli

У студии Ghibli свой особенный мультипликационный стиль рисования, с достаточно нежными цветами и оттенками. Я преобразовал в данный стиль фото Дженны Ортеги в образе Уэнсдей Аддамс (рис. 2.16).

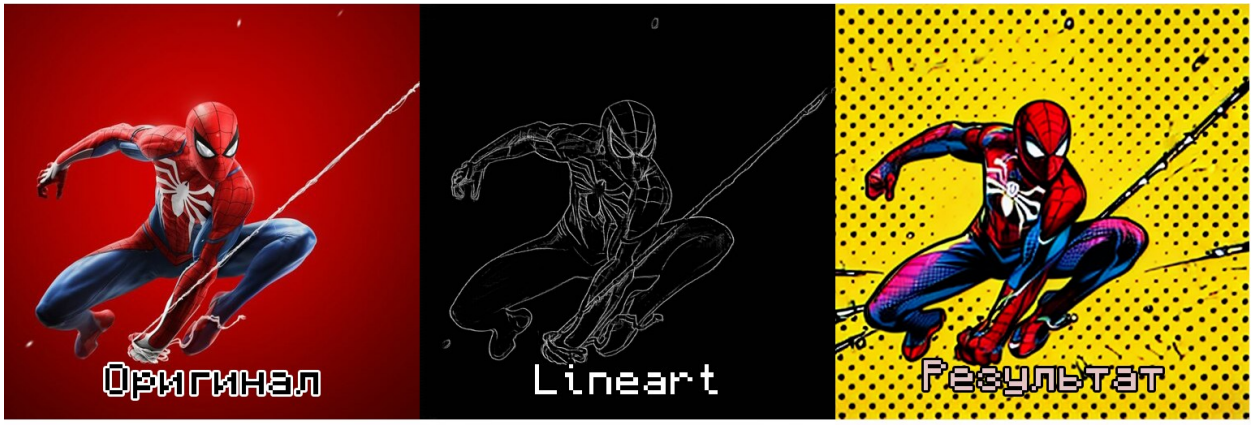
Позитивный промпт: ghibli style, 1girl, bangs, black eyes, black hair, looking at viewer, black dress, <lora:ghibli_style_offset:1>, masterpiece, <lora:detailed_eye_V2:0.1>, <lora:add_detail:0.5>
Негативный промпт: bad-artist, bad-artist-anime, easynegative, lr, badhandv4
Семплер: Heun Количество шагов: 20 CFG Scale: 7 Seed: 2128593574
Модель: FantasyWorldV1
Модуль ControlNet 1: препроцессор Lineart, модель control_v11p_sd15_lineart

Рис. 2.16. Демонстрационный набор 8 – Ghibli

Поп-арт комикс (рис. 2.17).

Рис. 2.17. Примеры работ в стиле поп-арт

Для стиля характерна техника в стиле коллажа, фотопечати, чередование разных текстур, яркие цвета – в этом стиле часто рисовали рекламу и комиксы. Я решил превратить в комиксный стиль изображение человека-паука из игры Marvel Spider-Man (рис. 2.18).



Позитивный промпт: pop art, spider man, (vivid yellow background:1.5), <lora:pop_art_v2:0.7>, white spider web, wrapped web around fist
Негативный промпт: bad-artist, (bad_prompt_version2:0.8), badhandv4, easynegative, ng_deepnegative_v1_75t, verybadimagenegative_v1.3
Семплер: DPM++SDE Karras Количество шагов: 22 CFG Scale: 7 Seed: 4036846171
Модель: AnythingV5Ink
Модуль ControlNet 1: препроцессор Lineart, модель control_v11p_sd15_lineart

Рис. 2.18. Демонстрационный набор 9 – Поп-арт

Dark art (рис. 2.19).

Рис. 2.19. Примеры Dark art

Работы Николы Самори это готические, мрачные картины в стиле барокко. Я попытался представить картину Шиканедера «Вечер в саду» в новом, пугающем свете (рис. 2.20).

Позитивный промпт: painting of a gloomy mansion at sunset, an abandoned garden, withered flowers the color of blood, a gloomy sinister figure in a cloak by a branching tree, pale moon, gothic terror, in the style of nicola samori, <lora:NicolaSamori:1.2>
Негативный промпт:
Семплер: DPM++2S a Karras Количество шагов: 25 CFG Scale: 6 Seed: 2596366099
Модель: sdvn7Nijistylexl_v1
Модуль ControlNet 1: препроцессор Canny, модель sai_xl_canny_256lora
Модуль ControlNet 2: препроцессор Depth, модель sai_xl_depth_256lora

Рис. 2.20. Демонстрационный набор 10 – Dark art

ЗАКЛЮЧЕНИЕ

Целью данной курсовой работы было создание демонстрационного набора стилизованных посредством нейросети изображений. В ходе работы все задачи, поставленные в начале, а именно анализ технологии, сбор и обработка изображений (10 шт.), были выполнены.

При создании коллекции была использована диффузионная нейросеть Stable Diffusion, а также рассмотрены все альтернативные способы для стилизации. Для анализа работы технологии переноса стиля на изображение были рассмотрены все встроенные возможности по влиянию на изображение и все актуальные модели ControlNet.

Во время работы над проектом были приобретены практические навыки в работе с нейросетью Stable Diffusion, сборе референсов, составлении промптов и стилизации изображений.

Подводя итоги проделанной в ходе курсовой работы, были достигнуты все цели и получены соответствующие знания. Далее наработанные при создании своей демонстрационной коллекции навыки можно будет использовать для свободного творчества или получения коммерческих материалов.

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. A. Dosovitskiy, P. Fischer, J. T. Springenberg, M. Riedmiller, T. Brox. "Discriminative Unsupervised Feature Learning with Exemplar Convolutional Neural Networks." arXiv:1406.6909, 2014.
2. F. Luan, S. Paris, E. Shechtman, K. Bala. “Deep Photo Style Transfer”. arXiv:1703.07511, 2017.
3. L. A. Gatys, A. S. Ecker, M. Bethge. " A Neural Algorithm of Artistic Style." arXiv:1508.06576, 2015.
4. L. A. Gatys, M. Bethge, A. Hertzmann, E. Shechtman. "Preserving Color in Neural Artistic Style Transfer." arXiv:1606.05897, 2016.
5. L. A. Gatys, A. S. Ecker, M. Bethge. "Image Style Transfer Using Convolutional Neural Networks." CVPR 2016.
6. L. Zhang, A. Rao, M. Agrawala. “Adding Conditional Control to Text-to-Image Diffusion Models”. arXiv:2302.05543, 2023.
7. J. Johnson, A. Alahi, L. Fei-Fei. "Perceptual Losses for Real-Time Style Transfer and Super-Resolution." arXiv:1603.08155, 2016.
8. M. Liu, T. Breuel, J. Kautz. "Unsupervised Image-to-Image Translation Networks." arXiv:1703.00848, 2017.
9. J. Ho, A. Jain, P. Abbeel, and A. A. Efros. "Stable Diffusion with Gradient Checkpointing". 2022
10. Raphael Tang, Linqing Liu, Akshat Pandey, et al. “What the DAAM: Interpreting Stable Diffusion Using Cross Attention”. arXiv:2210.04885, 2022.
11. Z. Wang, C. Xu, Z. Tu. "Learning to Adapt Structured Output Space for Semantic Segmentation." CVPR 2018.
12. T. Breuel, M. Liu, J. Hardeberg. "Stable Diffusion for Synchronizing Style Transfer." CVPR 2019.
13. T. Breuel, M. Liu. "Stable Diffusion for Synchronizing Neural Networks." ICML 2019.

14. J. Alammar. “The Illustrated Stable Diffusion” [сайт]. – 2022. – URL: http://jalammar.github.io/illustrated-stable-diffusion (дата обращения: 26.09.2023).

15. Stable Diffusion // Wikipedia: [сайт]. – 2023. – URL: https://en.wikipedia.org/wiki/Stable_Diffusion (дата обращения: 25.09.2023).

16. Ablation Study: Why ControlNets use deep encoder? // GitHub: [сайт]. – 2023. – URL: https://github.com/lllyasviel/ControlNet/discussions/188 (дата обращения: 11.10.2023).

17. ControlNet: Let us control diffusion models! // GitHub: [сайт]. – 2023. – URL: https://github.com/lllyasviel/ControlNet (дата обращения: 10.10.2023).

ПРИЛОЖЕНИЕ А

Руководство по установке расширений в Stable Diffusion

Чтобы установить любое расширение, необходимо проследовать следующей инструкции:

1. Открыть вкладку «Extensions»;
2. Перейти во вкладку «Available» и нажать на кнопку «Load From»;
3. В поисковую строку ниже ввести название нужного расширения;
4. В строке искомого расширения нажать на кнопку «Install»;
5. После установки необходимо перейти на вкладку «Installed» и последовательно нажать сначала на кнопку «Check for updates», а затем на «Apply and restart UI».

После полной установки расширения его можно найти на вкладке «Installed» и отключить при необходимости, либо нажать на ссылку в столбце «URL», чтобы перейти на страницу GitHub и узнать подробности о расширении.

ПРИЛОЖЕНИЕ Б

Руководство по установке сторонних материалов

Чтобы установить любую стороннюю модель, LoRA и прочее, необходимо перекинуть скачанный файл в папку, где расположена нейросеть, по следующему пути:

1. Для модели – файл с расширением «.safetensors» в папку «…\webui\models\Stable-diffusion»;
2. Для LoRA – файл с расширением «.safetensors» в папку «…\webui\models\Lora»;
3. Для эмбеддинга – файл с расширением «.pt» в папку «…\webui\embeddings»;
4. Для модели ControlNet – файл с расширением «.pth» в папку «…\webui\extensions\sd-webui-controlnet\models».

ПРИЛОЖЕНИЕ В

Ссылки на скачивание всех используемых материалов

Модели:

1. revAnimated_122EOL. URL: «https://civitai.com/models/7371/rev-animated»;
2. AnythingV5. URL: «https://civitai.work/models/9409?modelVersionId=90854»;
3. KanPiroMixV20. URL: «https://civitai.com/models/24591/kanpiromix»;
4. EpicRealism. URL: «https://civitai.com/models/25694?modelVersionId=160989»;
5. Dreamshaper8. URL: «https://civitai.com/models/4384?modelVersionId=128713»;
6. FantasyWorldV1. URL: «https://civitai.com/models/11031?modelVersionId=13069»;
7. Sdvn7Nijistylexl_v1. URL: «https://civitai.com/models/123307?modelVersionId=155870».

LoRA:

1. animeoutlinev4_16. URL: «https://civitai.com/models/16014?modelVersionId=28907»;
2. DiscoElysiumStyle. URL: «https://civitai.com/models/16048/or-disco-elysium-style-lora»;
3. Detailed_eye_V2. URL: «https://civitai.com/api/download/models/71804»;
4. Zyd232_InkStyle_v1. URL: «https://civitai.com/models/73305/zyd232s-ink-style»;
5. Zdzislaw_beksinski_finetuned. URL: «https://civitai.com/models/113803?modelVersionId=122978»;

6. Methurlant. URL: «https://civitai.com/models/83305?modelVersionId=88524»;

7. Ghibli_style_offset. URL: «https://civitai.com/models/6526?modelVersionId=7657»;

8. Add_detail. URL: «https://civitai.com/models/58390/detail-tweaker-lora-lora»;

9. Pop_art_v2. URL: «https://civitai.com/models/161450?modelVersionId=188417»;

10. NicolaSamori. URL: «https://civitai.com/models/161205?modelVersionId=181422».

Эмбеддинги:

1. easynegative. URL: «https://civitai.com/models/7808/easynegative»;

2. lr. URL: «https://civitai.com/models/43286/image-sharpener»;

3. bad-artist. URL: «https://huggingface.co/nick-x-hacker/bad-artist/tree/main»;

4. badhandv4. URL: «https://civitai.com/models/16993/badhandv4-animeillustdiffusion»;

5. bad_prompt_version_2. URL: « https://civitai.com/models/55700?modelVersionId=60095»;

6. ng_deepnegative_v1_75t. URL: «https://civitai.com/models/4629?modelVersionId=5637»;

7. verybadimagenegative_v1.3. URL: «https://civitai.com/models/11772?modelVersionId=25820»;

8. BadDream & UnrealisticDream. URL: «https://civitai.com/models/72437?modelVersionId=77169»;

9. FastNegativeV2. URL: «https://civitai.com/models/71961?modelVersionId=94057»;

10. advntr. URL: «https://civitai.com/models/6841?modelVersionId=8042».

Модели ControlNet:

1. Стандартные. URL: «https://huggingface.co/lllyasviel/ControlNet-v1-1/tree/main»;
2. T2I-Adapter. URL: «https://huggingface.co/TencentARC/T2I-Adapter/tree/main/models»;
3. XL. URL: «https://huggingface.co/lllyasviel/sd_control_collection/tree/main».