



Вклад участников проекта



Коновалов Михаил

- Подбор и настройка сервера для обучения
- Подготовка модели
- Эксперименты с зашумлением генератора и wasserstein loss
- Изучение статей с различными архитектурами
- Тестирование моделей на сервере



Смирнов Михаил

- Тестировал loss
- Для контентного лосса брал архитектуру VGG16_bn и рассматривал разные способы обрезать сеть, оставляя от 2 до 4 сверточных слоев слоев с возможностью дообучения и без.
- Для лосса генератора пробовал mse и bce(кросс энтропия).
- Делал финальные видео из пропущенных через сеть изображений
- Обучение модели на сервере, анализ составленных архитектур



Захватаев Михаил

- Нарезка и фильтрация (при помощи ядра Собеля) кадров выбранных мультфильмов для составления датасета
- Подготовка и тестирование на сервере новой архитектуры генератора и дискриминатора с различными лоссами
- Обучение модели без content loss
- Подготовка обработанных кадров для составления видео



Ссылки

- https://drive.google.com/drive/folders/1AiwOJpMDegT_ZC8ONAXo9dpWDh7G50RC - архивы лучших обученных моделей



Используемые ресурсы

- <https://github.com/junyanz/pytorch-CycleGAN-and-pix2pix> - Исходная реализация CycleGan на pytorch
- https://openaccess.thecvf.com/content_cvpr_2018/papers/Chen_CartoonGAN_Generative_Adversarial_CVPR_2018_paper.pdf - Статья с иной архитектурой генератора и дискриминатора
- <https://arxiv.org/pdf/1703.10593.pdf> - статья про Gan-ы
- <https://hardikbansal.github.io/CycleGANBlog/> - статья про Gan-ы
- <https://vast.ai/> - аренда GPU для более быстрого вычисления