



16 июня 2022 г.

# Колоризация изображений

Головин Вячеслав    Саленек Иван



## Постановка задачи

- Колоризация черно-белых изображений при помощи нейронных сетей
- Обзор существующих решений и попытка их улучшения



Проект DeOldify





В основе **DeOldify** лежит **GAN** архитектура. Автор использует собственный **NoGAN** подход, для обучения генератора и критика

Генераторам выступает **U-Net** из фреймворка `fast.ai`, позволяющего использовать предобученную сеть (н-р **ResNet**) в качестве основы для **U-Net**

**DeOldify** предлагает **2** предобученных "колорайзера":

- **Artistic** – `resnet34` в основе **U-Net** + 5 **NoGAN** итераций.
  - ✓ Более яркие и детальные результаты
  - ✗ Возникают артефакты
- **Stable** – `resnet101` в основе **U-Net** + 3 **NoGAN** итерации.
  - ✓ Лучше работает для пейзажей и портретов
  - ✗ Более тусклые цвета



## Реализованные бейзлайны

- В качестве базовых метрик были взяты **Feature Loss**, основанный на **VGG16**, и обычный **MSE**.
- Метрики были посчитаны для **1000** "колоризованных" картинок из **ImageNet** ([ссылка на папку с результатами](#))

Colorization Results on 1000 224x224 images				
Method	Model		Losses	
	Params (MB)	Runtime (s)	VGG16 (loss)	MSE (loss)
<b>CIC-ECCV16</b>	<b>129</b>	<b>250</b>	<b>.52579</b>	<b>.00902</b>
<b>CIC-siggraph17</b>	<b>137</b>	<b>1232</b>	<b>.45244</b>	<b>.00655</b>
<b>DeOldify Stable</b>	<b>874</b>	<b>4301</b>	<b>.45699</b>	<b>.00695</b>
<b>DeOldify Artist</b>	<b>255</b>	<b>4287</b>	<b>.46735</b>	<b>.00696</b>

Ошибки колоризации изображений



## Текущие результаты

- Подготовлен датасет (100 + 5 + 10 тыс. изображений из ImageNet) и инструменты для работы с ним в `pytorch`;
- Написан модуль `VGG16Loss` для расчёта **Perceptual** / Feature loss (фактически аналогично ДЗ по style transfer);
- Разработан и обучен первый вариант **U-Net** генератора на основе `resnet34`. При текущей реализации получаемые изображения имеют слабый контраст и серо-зеленый оттенок, простые изменения архитектуры **U-Net** (добавление простых ResNet-блоков и батч нормализации в декодер) и параметров perceptual loss не приводят к существенным изменениям.



Сгенерированное U-Net и исходное изображения



## План экспериментов

- Модификация архитектуры **U-Net**: использование более сложных энкодеров, добавления улучшений, использованных в **DeOldify** (self-attention, spectral normalization).
- Обучение в режиме **NoGAN**. *Опционально: решение проблемы автоматического поиска момента остановки обучения.*
- Эксперименты с loss-функцией, поиск оптимальных параметров для получения ярких изображений.



## Ссылки

- GitHub
- Google Drive