

Министерство науки и высшего образования Российской Федерации
ФГАОУ ВО «Южно-Уральский государственный университет (НИУ)»
Институт естественных и точных наук
Факультет математики, механики и компьютерных технологий
Кафедра прикладной математики и программирования
Направление подготовки: 01.03.02 «Прикладная математика и информатика»

СВЕРХРАЗРЕШЕНИЕ ИЗОБРАЖЕНИЙ С ПОМОЩЬЮ
ГЕНЕРАТИВНО-СОСТЯЗАТЕЛЬНОЙ НЕЙРОННОЙ СЕТИ

Руководитель:
доцент кафедры ПМиП,
к.ф.-м.н.
Т.В. Карпета

Автор работы:
студент группы ЕТ-412
М.А. Васюк

Челябинск, 2020



ЦЕЛЬ И ЗАДАЧИ

Целью данной работы является разработка искусственной нейронной сети для повышения разрешения изображений без потери качества.

Для достижения данной цели необходимо решить следующие задачи:

- изучение и исследование существующих методов для решения задачи повышения разрешения изображения без потери качества;
- сбор и подготовка данных для обучения нейронной сети;
- разработка математической модели искусственной нейронной сети;
- обучение сети и оценка качества модели на экспериментальных данных.



СВЕРХРАЗРЕШЕНИЕ ИЗОБРАЖЕНИЙ

X – множество изображений низкого разрешения.

Y – множество изображений высокого разрешения.

Необходимо восстановить целевую функцию $f^*: X \rightarrow Y$.

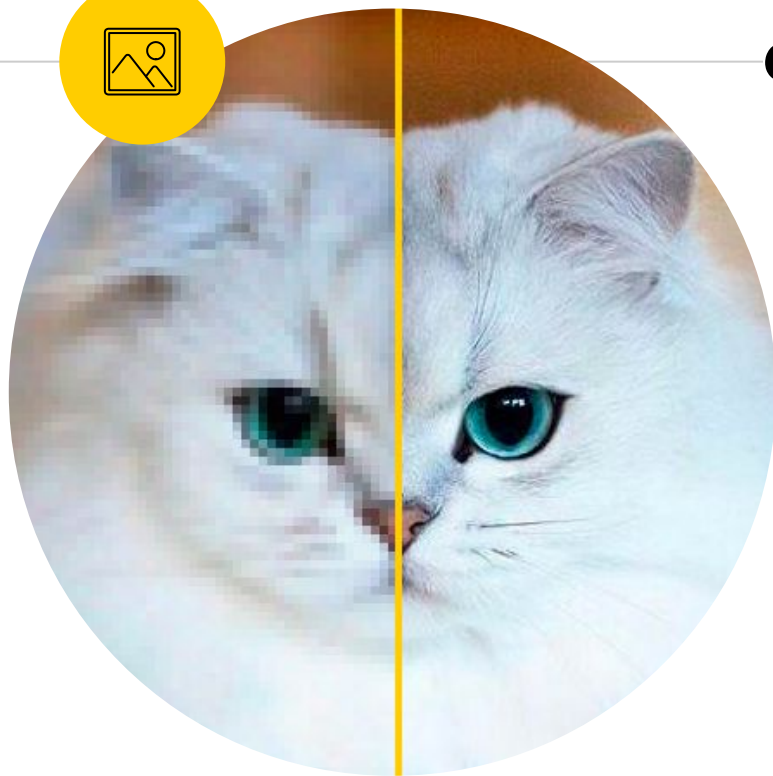


Рисунок 1 – Повышение разрешения изображения



СУЩЕСТВУЮЩИЕ МЕТОДЫ

Интерполяция

Нейронные
сети



СУЩЕСТВУЮЩИЕ РЕШЕНИЯ

- Bigjpg.com;
- Let's Enhance;
- Pichance.com;
- waifu2x.



Рисунок 2 – Пример работы
сервиса Let's Enhance



ИСХОДНЫЕ ДАННЫЕ

- DIV2K;
- Flickr1024;
- InStereo2K;
- ImageNet.



10 000 пар изображений:

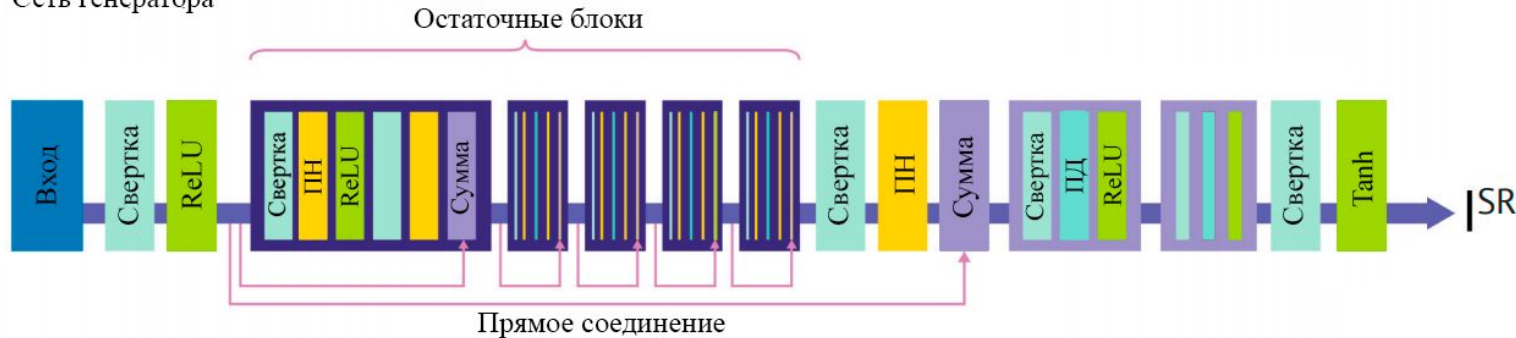
80% – обучающая,

20% – тестовая.



АРХИТЕКТУРА СЕТИ SRGAN

Сеть генератора



Сеть дискриминатора



ПН – пакетная нормализация, ПД – слой повышения дискретизации, ReLU, Leaky ReLU, Tanh, Sigmoid – функции потерь, HR – изображение высокого разрешения, SR – сгенерированное изображение (выход сети)

Рисунок 3 – Архитектура сети



ФУНКЦИИ АКТИВАЦИИ

Leaky ReLU:

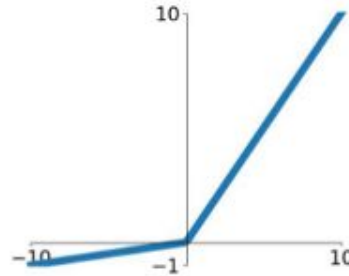


Рисунок 4 – Функция активации Leaky ReLU

$$f(y) = \begin{cases} 0.01y, & y < 0, \\ y, & y > 0. \end{cases} \quad (1)$$

Sigmoid:

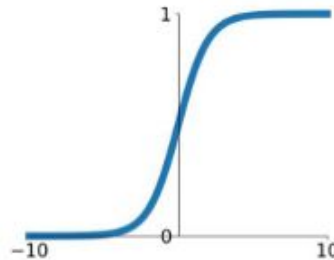


Рисунок 5 – Функция активации Sigmoid

$$f(y) = \frac{2}{1 + e^{-2y}} - 1. \quad (2)$$



ФУНКЦИИ ПОТЕРЬ

$$l_{CL} = \frac{1}{mn} \sum_{i=1}^m \sum_{j=1}^n (I_{i,j}^{HR} - G(I^{LR})_{i,j})^2, \quad (3)$$

$$l_{AL} = - \sum_{i=1}^N \ln D(G(I_i^{LR})), \quad (4)$$

m – ширина изображения,

n – высота изображения,

i, j – координаты пикселя,

I^{HR} – эталонное изображение высокого разрешения,

I^{LR} – изображение низкого разрешения,

$G(I^{LR})$ – сгенерированное сетью изображение высокого разрешения,

$D(G(I^{LR}))$ – выход дискриминатора для сгенерированного изображения,

N – количество изображений в пакете.



ФУНКЦИИ ПОТЕРЬ

$$l = l_{CL} + 0.001 \cdot l_{AL}, \quad (5)$$

l_{CL} – функция потерь контента,

l_{AL} – состязательная функция потерь.

$$MSE = \frac{1}{N} \sum_{i=1}^N (p_i - y_i)^2, \quad (6)$$

N – количество изображений в пакете,

p_i – реальный выход дискриминатора для i -го изображения,

y_i – верный (желаемый) выход дискриминатора.



СХЕМЫ АЛГОРИТМОВ



Рисунок 6 – Алгоритм работы программы



СХЕМЫ АЛГОРИТМОВ

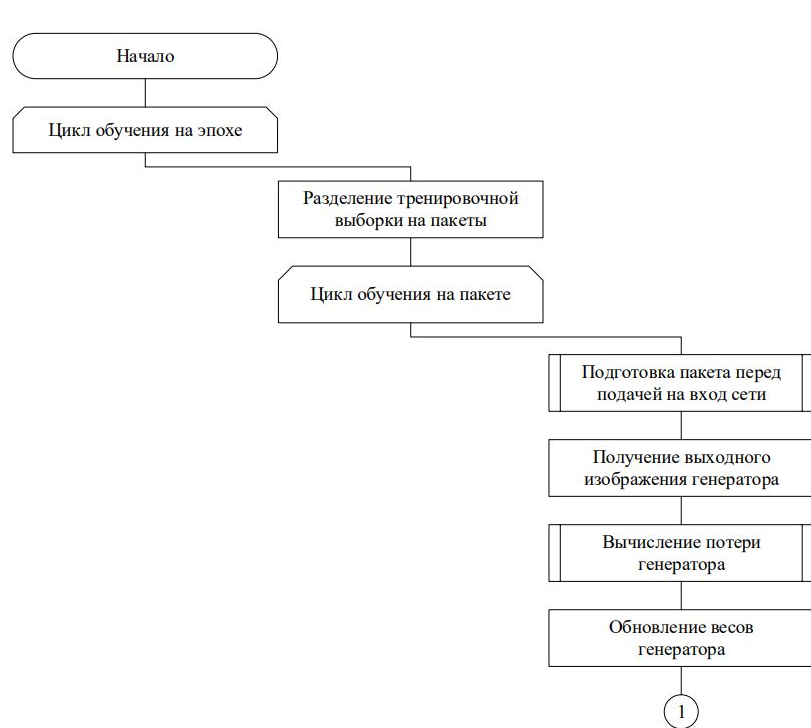


Рисунок 7 – Алгоритм обучения сети (1)

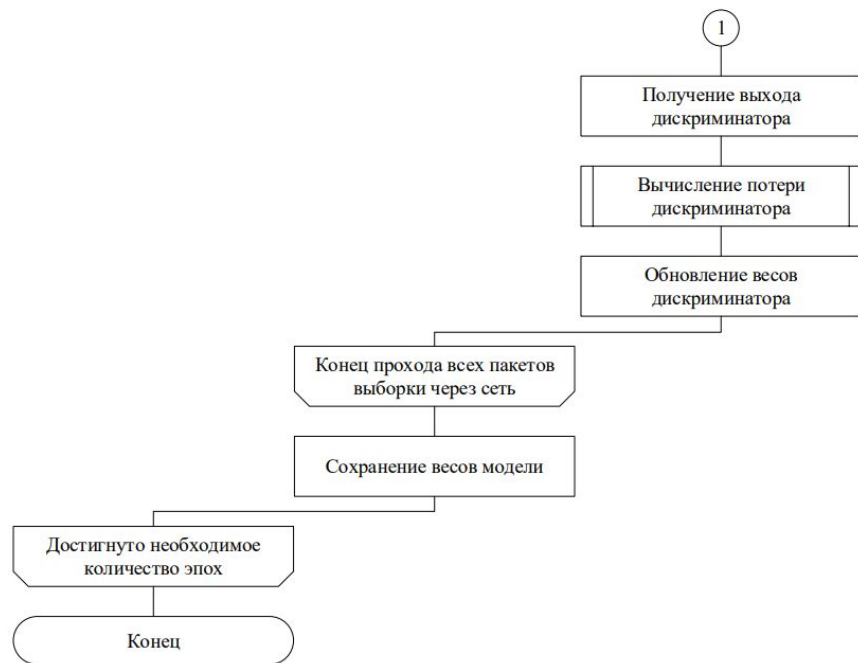


Рисунок 8 – Алгоритм обучения сети (2)



РЕЗУЛЬТАТЫ ОБУЧЕНИЯ



Рисунок 9 – График обучения дискриминатора

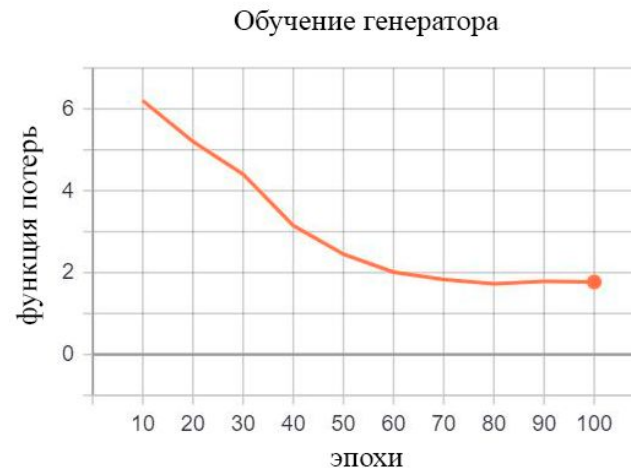


Рисунок 10 – График обучения генератора



МЕТРИКИ КАЧЕСТВА

$$PSNR = 20 \lg \left(\frac{255}{\sqrt{\sigma}} \right), \quad (7)$$

σ – среднеквадратичное отклонение для сравниваемого (x) и эталонного (y) изображений:

$$\sigma = \frac{1}{3mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} \sum_{l=0}^2 |x_l(i, j) - y_l(i, j)|^2, \quad (8)$$

- m – ширина изображения,
- n – высота изображения,
- l – номер канала изображения,
- i, j – координаты пикселя.



МЕТРИКИ КАЧЕСТВА

$$SSIM = \frac{(2\mu_x\mu_y + K)(2\sigma_{xy} + K)}{(\mu_x^2 + \mu_y^2 + K)(\sigma_x^2 + \sigma_y^2 + K)}, \quad (9)$$

- x – сравниваемое изображение,
- y – эталонное изображение,
- μ_x – среднее арифметическое сравниваемого изображения,
- μ_y – среднее арифметическое эталонного изображения,
- σ_x^2 – дисперсия для сравниваемого изображения,
- σ_y^2 – дисперсия для эталонного изображения,
- σ_{xy} – корреляционный момент изображений,
- K – выравнивающий коэффициент (равен 0.01).



ПРИМЕРЫ РАБОТЫ СЕТИ

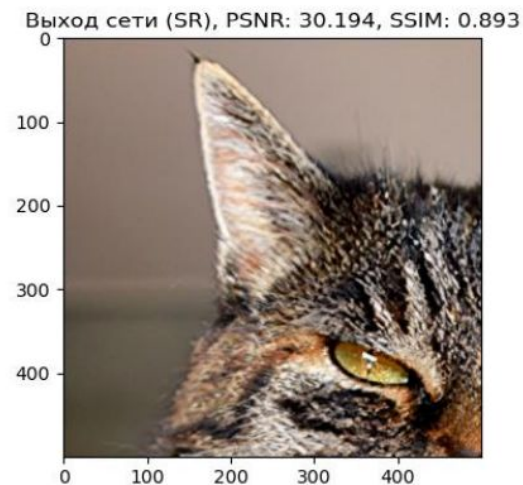
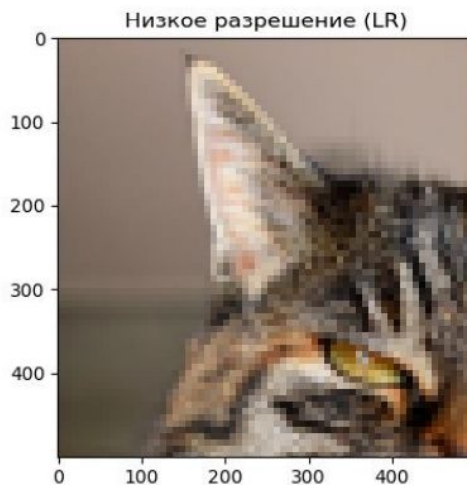
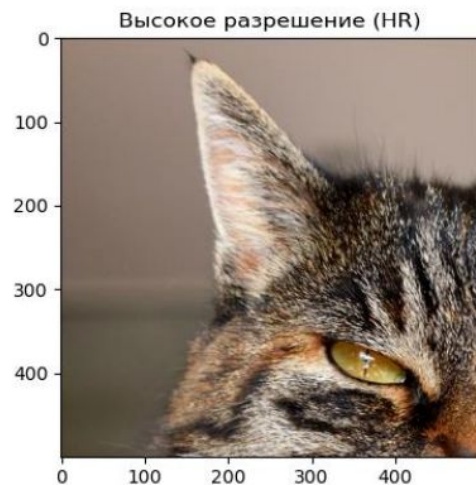


Рисунок 11 – Пример работы сети



ПРИМЕРЫ РАБОТЫ СЕТИ

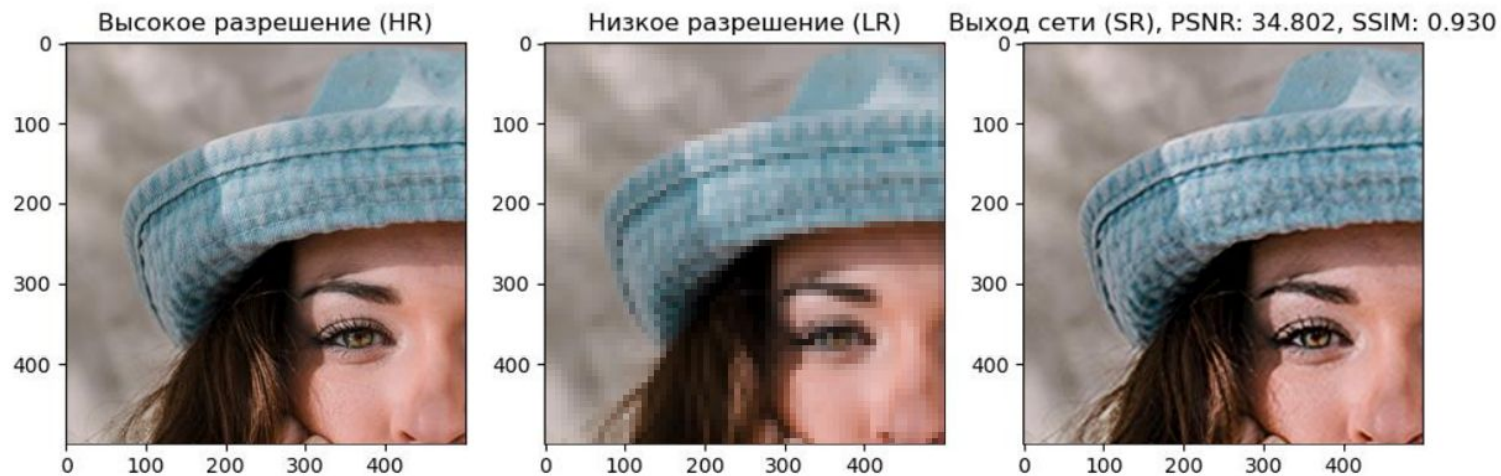


Рисунок 12 – Пример работы сети



СРАВНИТЕЛЬНЫЙ АНАЛИЗ

Таблица 1 – Сравнение с интерполяционными методами

	Ближайший сосед	Билинейная интерполяция	Бикубическая интерполяция	Интерполяция Ланцоша	GAN
PSNR	29.40	29.84	30.64	30.51	32.05
SSIM	0.79	0.84	0.87	0.88	0.91



ЗАКЛЮЧЕНИЕ

- в результате работы, было проведено исследование существующих методов решения задачи сверхразрешения;
- для обучения нейронной сети был собран и подготовлен набор данных изображений;
- была построена математическая модель нейронной сети, а также описаны методы ее обучения;
- данная сеть была реализована, обучена и протестирована на тестовых данных;
- проведен сравнительный анализ эффективности методов повышения разрешения изображений.

Полученное решение имеет практическое применение в проектах компании ООО «Цифровая собственность».

СПАСИБО ЗА ВНИМАНИЕ!



ФОРМУЛЫ

$$\mu_x = \frac{1}{M} \sum_{i=1}^M x_i; \quad (2.20)$$

где σ_x^2 – дисперсия для оригинального изображения (σ_y^2 – для оцениваемого) размером M пикселей:

$$\sigma_x^2 = \frac{1}{M} \sum_{i=1}^M (x_i - \mu_x)^2; \quad (2.21)$$

где σ_{xy} – корреляционный момент участков сравниваемых изображений:

$$\sigma_{xy} = \frac{1}{M} \sum_{i=1}^M (x_i - \mu_x)(y_i - \mu_y); \quad (2.22)$$



ФОРМУЛЫ

Алгоритм работы:

1) инициализация начальных моментов: $M_0 = 0, R_0 = 0$;

2) для каждого веса $W_t, t = 1, \dots, T$ выполнить:

– оценку первого момента:

$$M_t = \beta_1 M_{t-1} + (1 - \beta_1) \nabla L_t(W_{t-1}); \quad (2.12)$$

– оценку второго момента:

$$R_t = \beta_2 R_{t-1} + (1 - \beta_2) \nabla L_t(W_{t-1})^2; \quad (2.13)$$

– коррекцию смещения первого момента:

$$\widehat{M}_t = \frac{M_t}{(1 - \beta_1)^t}; \quad (2.14)$$

– коррекцию смещения второго момента:

$$\widehat{R}_t = \frac{R_t}{(1 - \beta_2)^t}; \quad (2.15)$$

– обновление:

$$W_t = W_{t-1} - \alpha \frac{\widehat{M}_t}{\sqrt{\widehat{R}_t} + \varepsilon} \quad (2.16)$$

3) вернуть веса W_T .



ФОРМУЛЫ

Шаг 1. Инициализация синаптических весов и смещений. В циклах по $k = 1, 2, \dots, K; i = 1, 2, \dots, H_k; j = 0, 1, 2, \dots, H_{k-1}$ генератор случайных чисел присваивает синаптическим весам и смещениям $w_{ij}^{(k)}$ малые величины, например, из диапазона от -1 до 1.

Шаг 2. Открытие цикла по $q = 1, 2, \dots, Q$. Представление из обучающего множества примеров очередного входного вектора $X_q = (x_1, x_2, \dots, x_N)_q$ и соответствующего ему желаемого выходного вектора $D_q = (d_1, d_2, \dots, d_M)_q$, где q – номер примера в обучающем множестве.



ФОРМУЛЫ

Шаг 3. Прямой проход. В циклах по $k = 1, 2, \dots, K; i = 1, 2, \dots, H_k$ вычисляются выходные сигналы i -го нейрона в k -ом слое:

$$y_i^{(k)} = f \left(\sum_{j=0}^{H_{k-1}} w_{ij}^{(k)} y_j^{(k-1)} \right), \quad (2.7)$$

где $y_j^{(0)} = x_j; x_0 = 1; y_0^{(k-1)} = 1; y_i = y_i^{(K)}$ – выходные сигналы персептрона.

Шаг 4. Обратный проход. В циклах по $k = K, K - 1, \dots, 1; i = 1, 2, \dots, H_k; j = 0, 1, 2, \dots, H_{k-1}$ вычисляются синаптические веса на новой эпохе:

$$w_{ij}^{(k)}(t+1) = w_{ij}^{(k)}(t) + \Delta w_{ij}^{(k)}, \quad (2.8)$$

$$\Delta w_{ij}^{(k)} = \eta \delta_i^{(k)} y_j^{(k+1)}. \quad (2.9)$$

Для нейронов выходного слоя $k = K$:

$$\delta_i^{(K)} = y_i(1 - y_i)(d_i - y_i). \quad (2.10)$$

Для остальных скрытых слоев:

$$\delta_i^{(k)} = y_i^{(k)} \left(1 - y_i^{(k)} \right) \sum_{l=1}^{H_{k+1}} \delta_l^{(k+1)} w_{li}^{(k+1)}. \quad (2.11)$$

Шаг 5. Закрытие цикла по q .

Шаг 6. Повторение шагов 2–5 необходимое количество раз. Векторы обучающих примеров X_q и D_q на шаге 2 алгоритма обычно представляются последовательно от первого до последнего, т. е. $q = 1, 2, \dots, Q$, где Q – общее количество примеров.