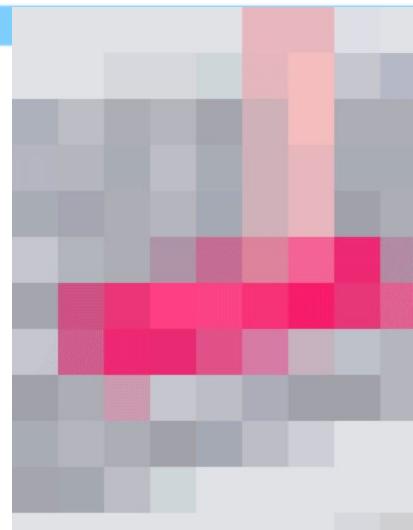
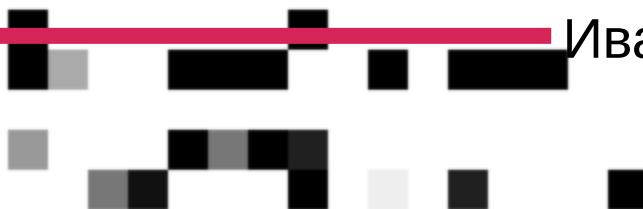




Иван Молодецких



Содержание

- Введение
 - Актуальность
 - Лол! Просто увеличь это!
 - Так ещё мой отец и мой дед делали
 - Смартфоны — двигатель прогресса
 - Примеры нейросетевых архитектур
 - Лаборатория и SR

Типичное обращение

Сильно сжатое видео



«У нас есть видео с подозреваемым,
пожалуйста, помогите восстановить лицо»

«Здесь не видно рук человека,
пожалуйста, помогите увеличить»

Типичное обращение

Низкая освещённость



«Помогите увеличить номер с камеры видеонаблюдения»

Типичное обращение

Письмо из полиции Англии

Use of commercial filters

1 письмо

<*****@*****.fsnet.co.uk>

Кому: <***@graphics.cs.msu.ru>

I have used your filters privately for some time to rescue my poor videos of family holidays but I would like to use the commercial filters for my work. I am currently a Police Officer in a small police force and we are getting a lot of CCTV video, which sometime is very poor quality and I can see how your filters would make a real difference. Can you tell me the cost and if I could use them.

Thank you

Использование коммерческих фильтров

1 письмо

<*****@*****.fsnet.co.uk>

Кому: <***@graphics.cs.msu.ru>

Я уже пользовался вашими фильтрами для личных целей, чтобы спасти свои плохие видео с семейных праздников. Но мне бы хотелось использовать коммерческие фильтры в своей работе. В данный момент я офицер полиции в небольшом подразделении. Мы получаем большое количество видео с камер видеонаблюдения, иногда очень плохого качества, и ваши фильтры действительно помогут. Не могли бы вы сказать мне их стоимость, и могу ли я их использовать?

Спасибо

Содержание

- Введение
 - Актуальность
 - **Лол! Просто увеличь это!**
- Так ещё мой отец и мой дед делали
- Смартфоны — двигатель прогресса
- Примеры нейросетевых архитектур
- Лаборатория и SR

«STOP!» and «Enhance it!»



«STOP!» and «Enhance it!»



Выводы

- Задача повышения разрешения была, есть и будет актуальна
- В массовом сознании заезженный штамп, не имеющий ничего общего с реальностью

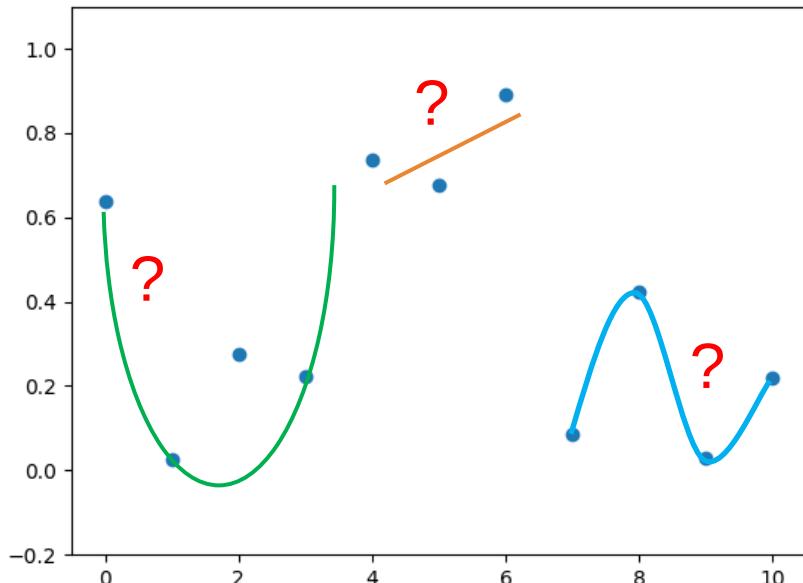
Содержание

- Введение
- **Так ёщё мой отец и мой дед делали**
- Смартфоны — двигатель прогресса
- Примеры нейросетевых архитектур
- Лаборатория и SR

Интерполяция

Постановка задачи

- Найти промежуточные значения по известному набору существующих точек
- Построить функцию, которая наилучшим образом аппроксимирует конечный набор заданных точек



- В общем, задача плохо обусловлена
- Выбор подходящего метода зависит от природы исходных данных

Интерполяция

Повышение разрешения

- Процесс интерполяции пытается восстановить недостающую информацию на основе предполагаемой зависимости между соответствующими изображениями низкого и высокого разрешения

A	B	C
D	E	F
G	H	I



A	?	B	?	C
?	?	?	?	?
D	?	E	?	F
?	?	?	?	?
G	?	H	?	I

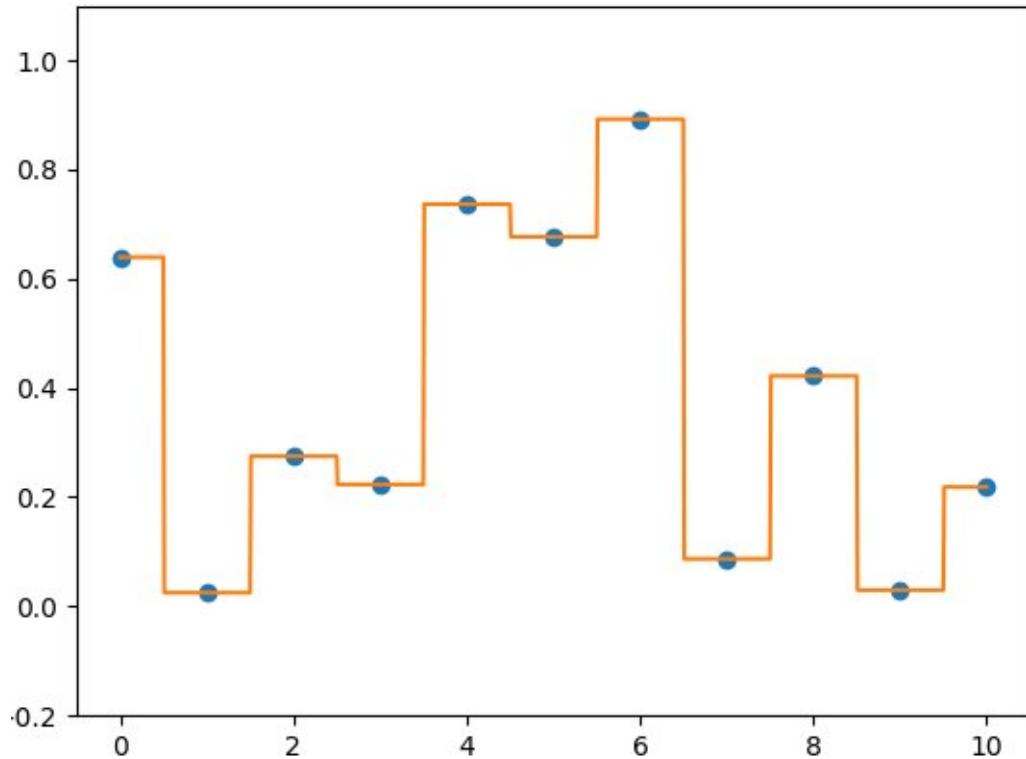
Низкое разрешение

Высокое разрешение

Одномерный случай

Метод ближайшего соседа

Значение в неизвестной точке совпадает со значением ближайшего соседа из исходного набора точек

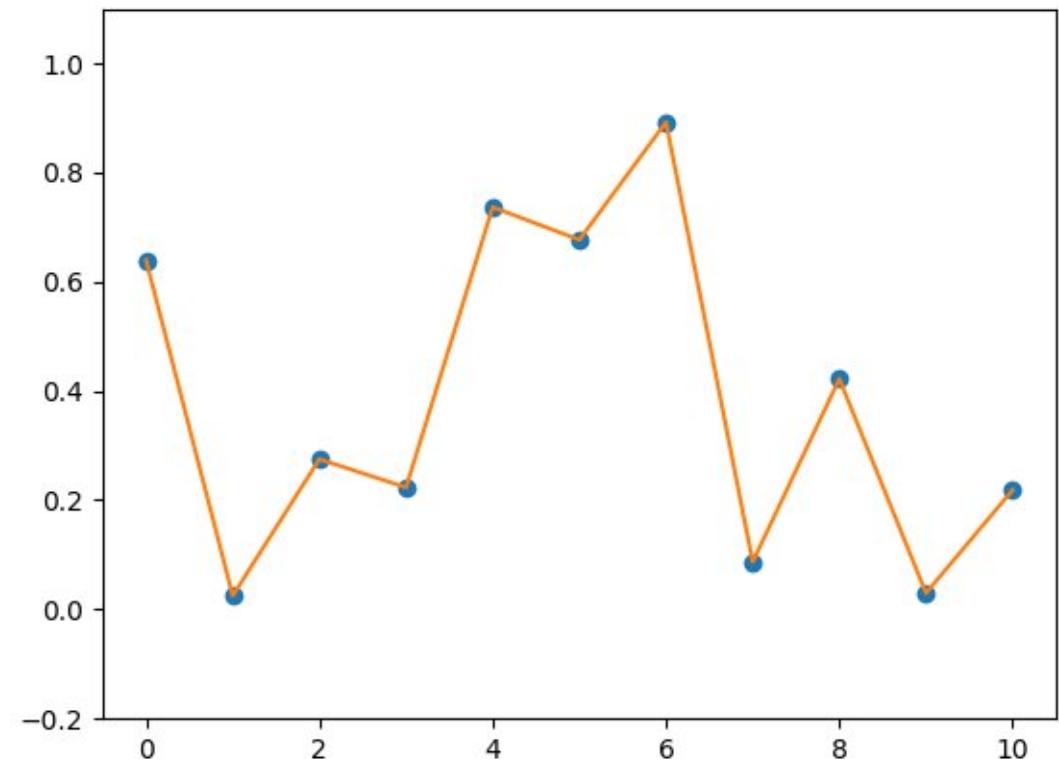


Простой в реализации
и вычислительно
эффективный
Низкое качество
интерполяции, особенно
высокочастотного сигнала

Одномерный случай

Линейная интерполяция

Приблизим значения в неизвестных областях по 2 соседним точкам с помощью кусочно-линейной функции.



Тогда значение в точке x на отрезке $[x_0, x_1]$ вычисляется по формуле:

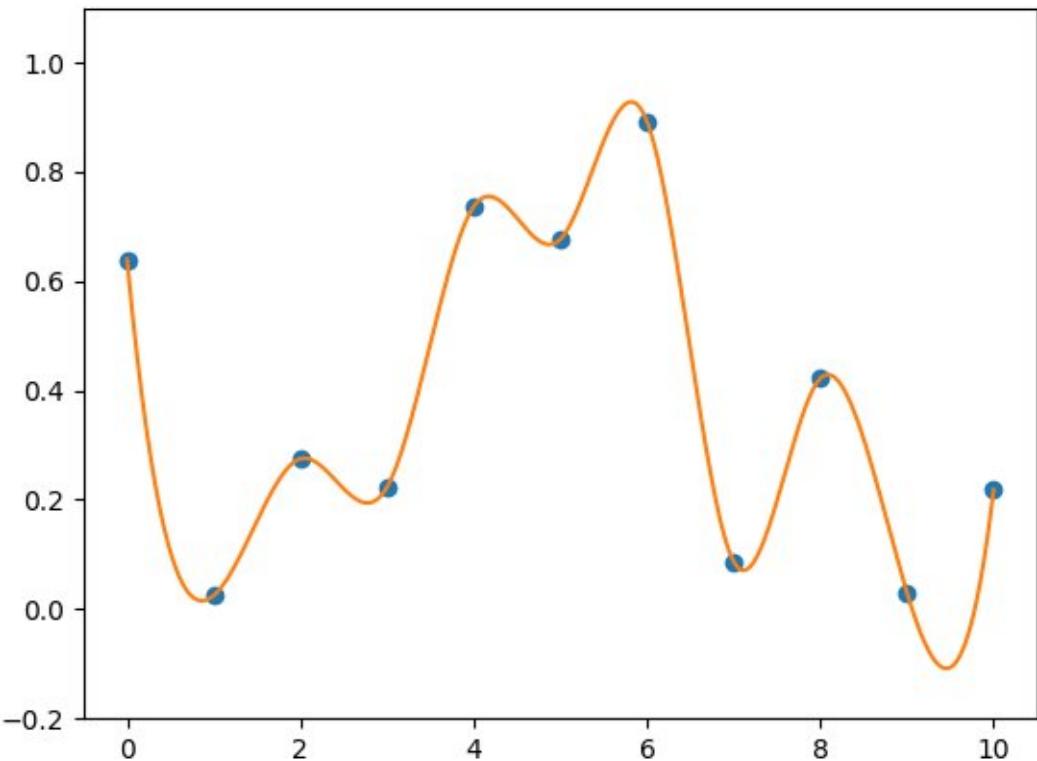
$$f(x) = \frac{(x-x_0)f(x_0)+(x_1-x)f(x_1)}{x_1 - x_0}$$

Получаем более гладкую аппроксимацию исходных точек

Одномерный случай

Кубическая интерполяция

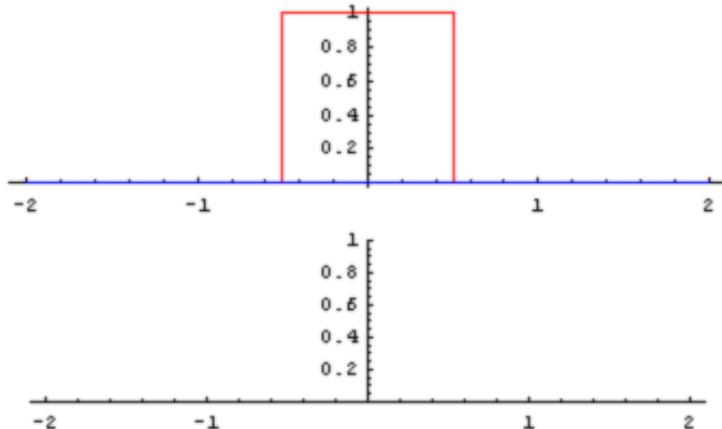
Приблизим значения в неизвестных областях по 4 соседним точкам с помощью полинома: $ax^3 + bx^2 + cx + d$



Коэффициенты находят по значениям в 4 точках (кубическая и.), либо по значениям и производным в 2 точках (и. кубическими сплайнами) За производную принимается уклон линии между предыдущей и следующей точкой

Одномерный случай

Обобщение через свёртки (1)



$$a(n) = [2, 1, 3, -1, 0, 0]$$

2	1	3	-1	0	0
---	---	---	----	---	---

$$b(n) = [-1, 1, 2, 0, 0, 0]$$

-1	1	2	0	0	0
----	---	---	---	---	---

отразить

0	0	0	2	1	-1
---	---	---	---	---	----

$a(m)$	2	1	3	-1	0	0
$b(0-m)$	-1	0	0	0	2	1
$b(1-m)$	1	-1	0	0	0	2
$b(2-m)$	2	1	-1	0	0	0
$b(3-m)$	0	2	1	-1	0	0
$b(4-m)$	0	0	2	1	-1	0
$b(5-m)$	0	0	0	2	1	-1

Свёртка функций в функциональном анализе — это операция, показывающая «схожесть» одной функции с отражённой и сдвинутой копией другой.

Свёртка последовательностей — это результат перемножения элементов двух заданных числовых последовательностей таким образом, что члены одной последовательности берутся с возрастанием индексов, а члены другой — с убыванием.

Одномерный случай

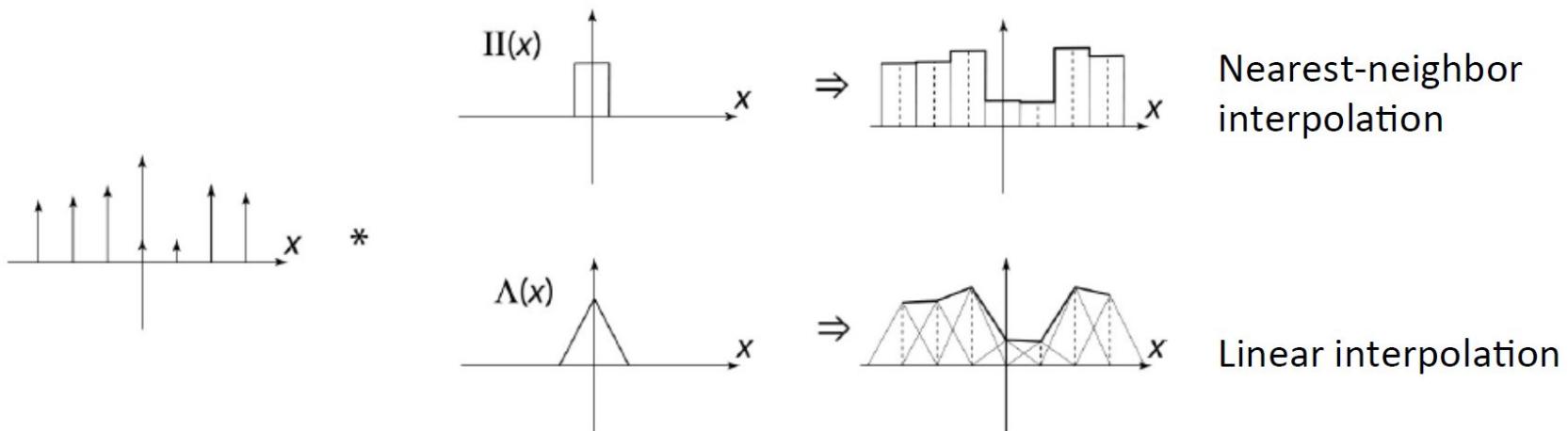
Обобщение через свёртки (2)

Предложенные функции можно записать в общем виде:

$$\hat{f}(x) = \sum_{k=-\infty}^{k=+\infty} c(x_k) \beta(x - x_k),$$

где $\beta(x)$ – функция и., $c(x)$ – коэффициенты и.

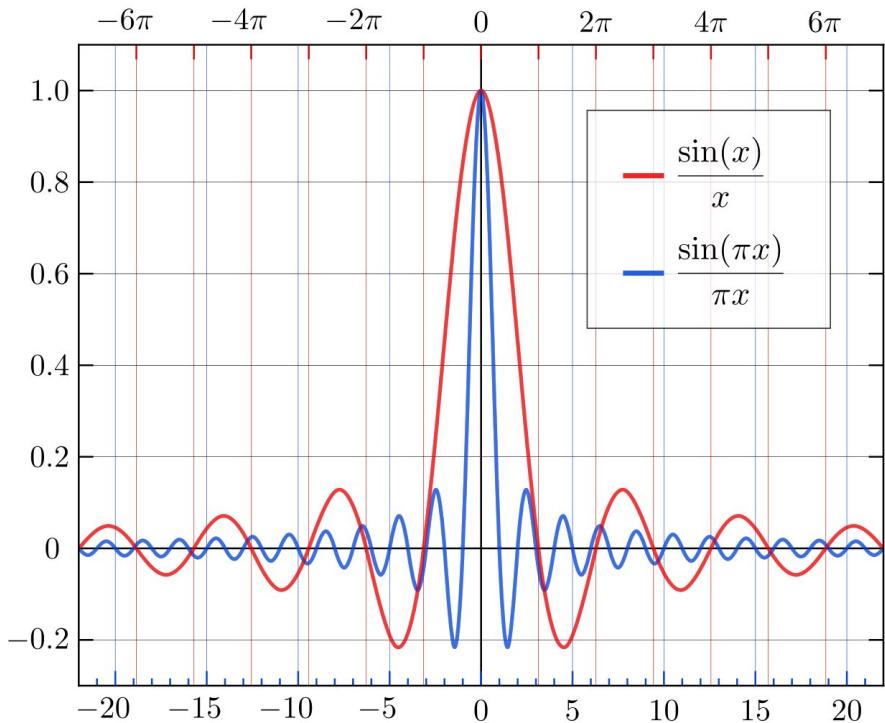
Это соответствует свёртке исходных точек с функцией интерполяции $\beta(x)$ с некоторыми коэффициентами $c(x)$



Одномерный случай

Sinc

Если искомая функция имеет ограниченную частоту (не имеет точек разрыва), то по теореме Котельникова она может быть восстановлена с какой угодно точностью по своим дискретным отсчётам, взятым с частотой $f > 2f_{max}$, где f_{max} – максимальная частота функции f



Для идеального восстановления используются свёртка с sinc функцией во временной области:

$$\text{sinc}_T = \frac{\sin(T * \pi x)}{T * \pi x}$$

выбранные точки

$$\tilde{f} = (f(x) H_T(x)) \otimes \text{sinc}_T(x)$$

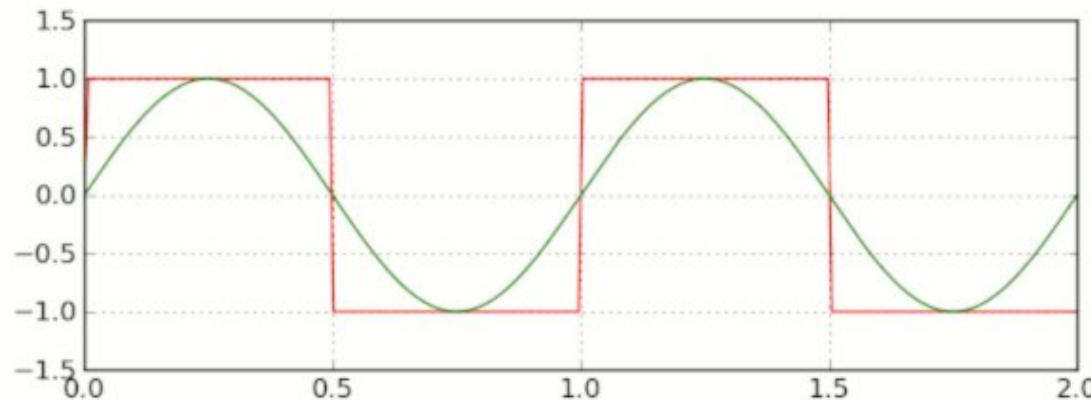
восстановленная функция

Одномерный случай

Недостатки sinc-функции



- На практике сигнал может прерываться или содержать резкие перепады и, как следствие, неограничен в частотной области
- Sinc-функция имеет бесконечную ширину, поэтому свёртку с ней можно вычислить лишь приближённо



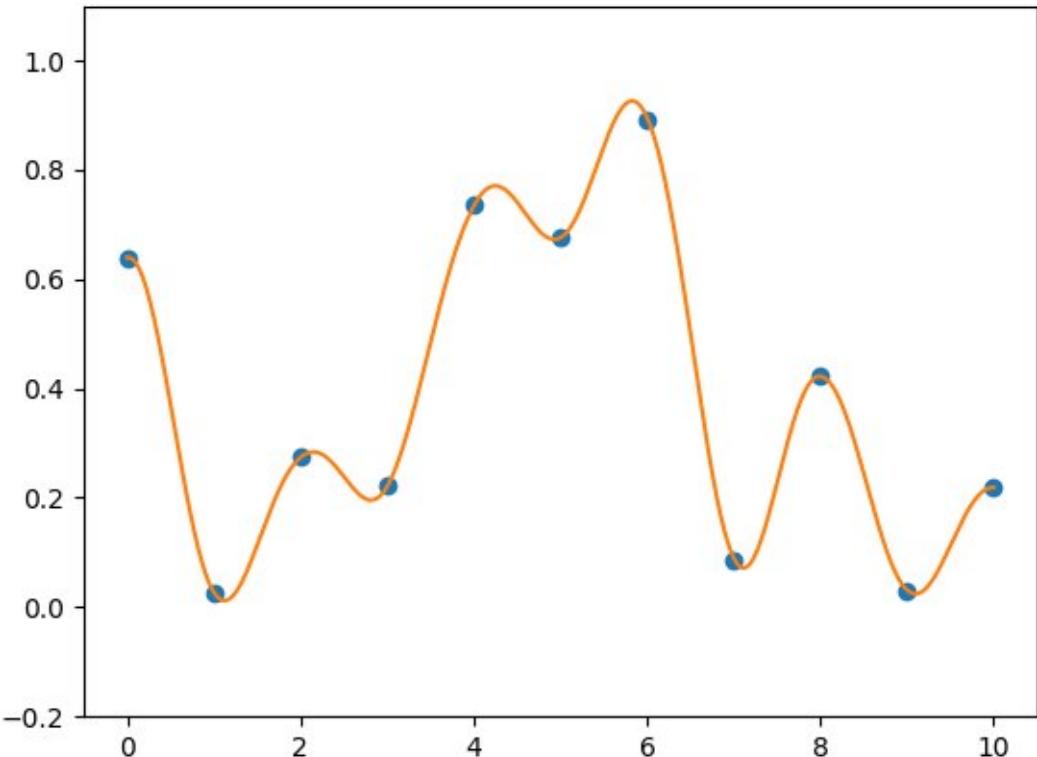
В результате, при применении идеального фильтра восстановления на границах возникает **эффект Гиббса**. Причём, чем больше гармоник участвуют в приближении, тем эффект проявляется сильнее

Одномерный случай

Фильтр Ланцоша

На практике используется фильтр Ланцоша (Lanczos) — модификация sinc-функции с ограниченной шириной (определяется параметром a):

$$L(x) = \text{sinc}(x) \text{sinc}(x/a), \text{ для } -a < x < a$$



- Показывает лучший результат восстановления для непрерывных сигналов
- Меньше страдает от эффекта Гиббса, чем sinc-функция
- В OpenCV, Avisynth для изображений используется вариант с радиусом $a = 4$ (окно 8×8 пикселей)

Артефакты на цифровых изображениях

Алиасинг (aliasing)

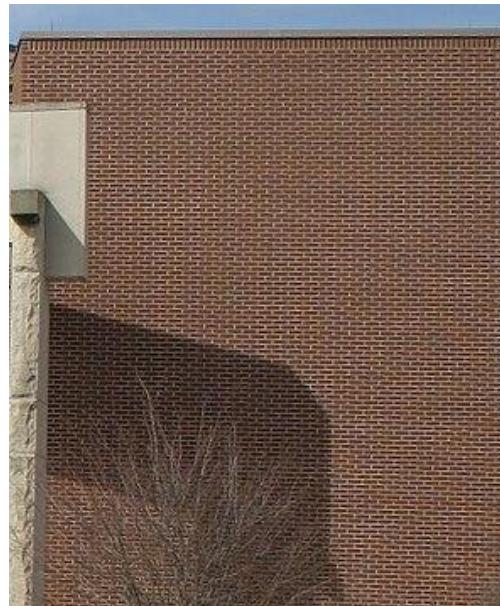


Эффект блочности при повышении разрешения. Эффект муара при сжатии. В общем случае, алиасинг означает маскирование высоких частот под низкие.



Исходное

Метод ближайшего
соседа $\times 4$



Исходное



Сжатое JPEG

Артефакты на цифровых изображениях

Размытие (blurring)



Визуально наиболее предпочтительный, вследствие чего билинейная и бикубическая и. чаще всего используются в качестве базовых методов.



Graphics & Media Lab
Video Group

Исходное

Метод ближайшего
соседа $\times 4$



Graphics & Media Lab
Video Group

Билинейная
интерполяция $\times 4$



Graphics & Media Lab
Video Group

Бикубическая
интерполяция $\times 4$

Артефакты на цифровых изображениях

Рингинг (ringing)



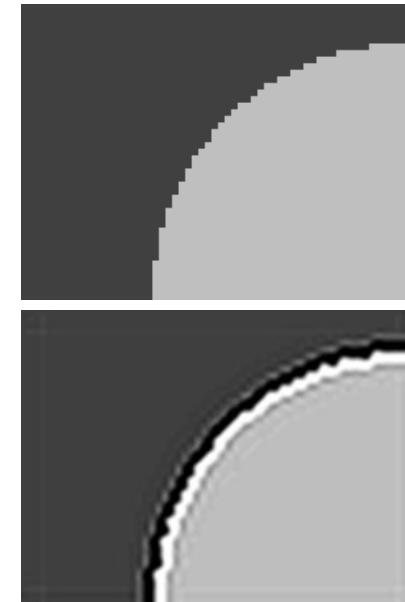
Фильтр Ланцоша выдаёт более резкое изображение, но страдает от рингинга. Все границы изображения будто дополнительно обведены



Исходное



Фильтр Ланцоша × 4



Отчетливый рингинг
на границах объектов

Выводы

- Теоретические исследования в области семплирования признают, что идеальное восстановление на практике недостижимо
- Поэтому любая интерполяция — это додумывание неизвестной информации
- Больше о математической стороне вопроса можно почитать [здесь](#) и [здесь](#)

Содержание

- Введение
- Так ёщё мой отец и мой дед делали
- **Смартфоны — двигатель прогресса**
- Примеры нейросетевых архитектур
- Лаборатория и SR

Google Pixel Camera

Эволюция



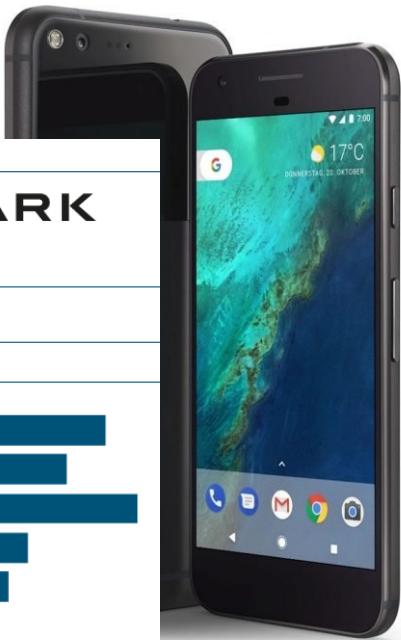
Google Pixel



DXOMARK
CAMERA

90 PHOTO

Exposure:	90
Color:	80
Autofocus:	98
Texture:	70
Noise:	65
Artifacts:	50
Night:	41
Zoom:	24
Bokeh:	30
Wide:	N/A



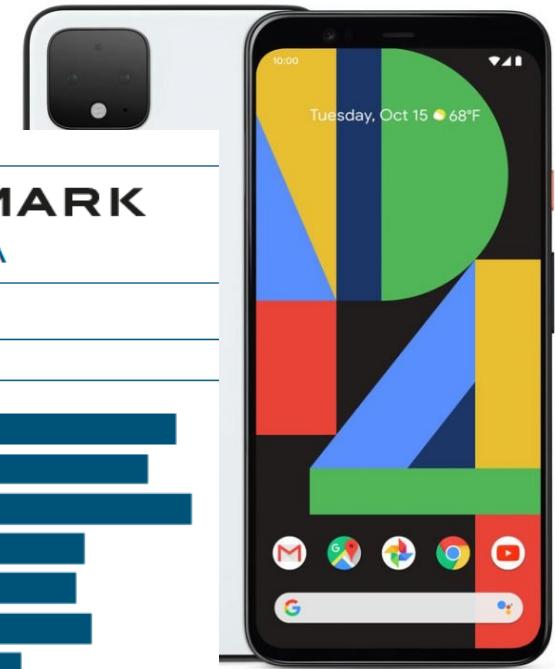
Google Pixel 4



DXOMARK
CAMERA

117 PHOTO

Exposure:	95
Color:	88
Autofocus:	99
Texture:	72
Noise:	70
Artifacts:	74
Night:	56
Zoom:	81
Bokeh:	60
Wide:	N/A



За 3 года за счёт алгоритмов повышения разрешения и дополнительных камер удалось на порядок увеличить качество цифрового зума

RAISR (Pixel 2)

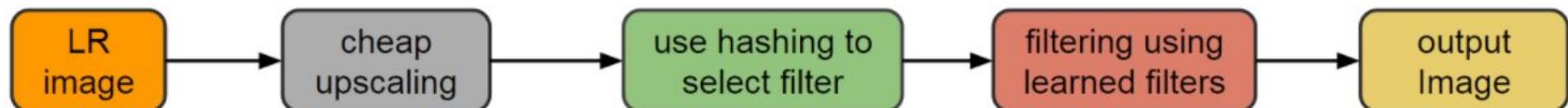
Основная идея

Использовать набор предобученных (ML) фильтров для разных участков изображения, вместо фиксированного линейного фильтра



Работа алгоритма

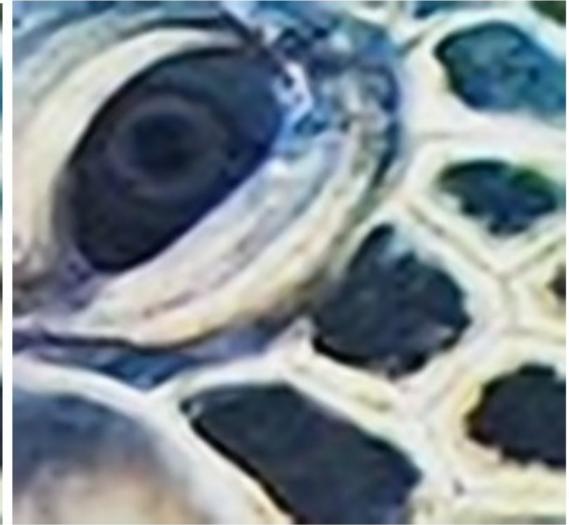
Для начала изображение увеличивается с помощью билинейной интерполяции, далее считаются характеристики каждой области (наличие границ, текстур) и на их основе алгоритм выбирает фильтр из предобученного множества



Низкое разрешение



Bicubic × 2



RAISR × 2

RAISR

Примеры



Сверху: исходное изображение
Снизу : RAISR × 2



Слева : исходное изображение
Справа: RAISR × 3





Выше PSNR



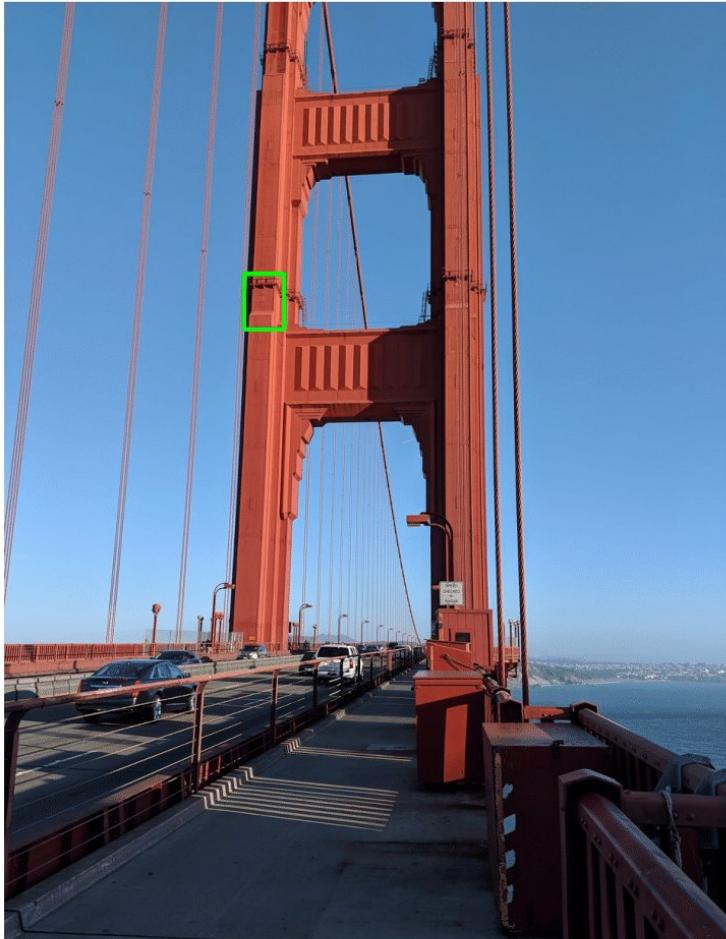
Визуально лучше



Качество текущих алгоритмов оцениваются с помощью PSNR. В результате, методы стремятся получить чересчур сглаженные изображения без текстур (высокочастотный сигнал), которые выглядят неестественно

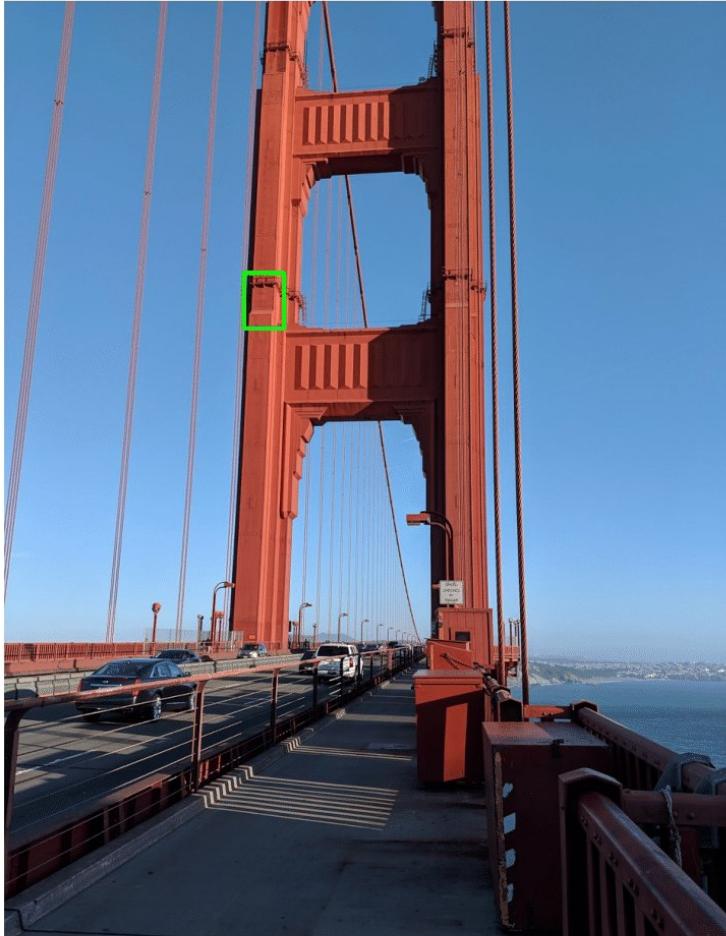
Super Res Zoom

Pixel 2 vs Pixel 3 (1)



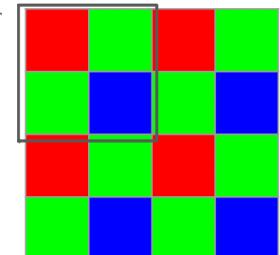
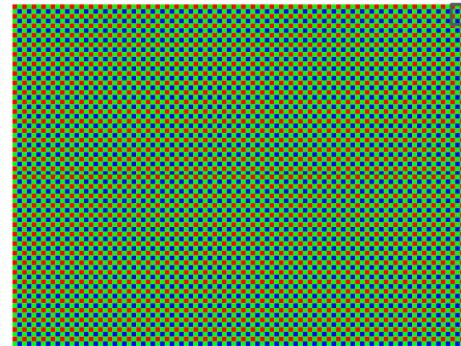
Super Res Zoom

Pixel 2 vs Pixel 3 (2)



Super Res Zoom

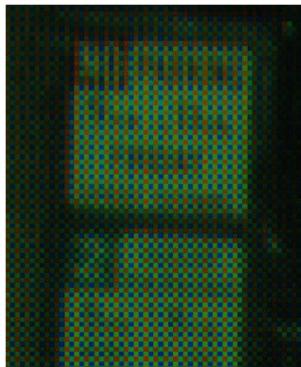
Съёмка со смартфона



Bayer mosaic

Размер матрицы смартфона и зеркальной камеры

Байеровский шаблон



RAW image



Demosaic



Denoise



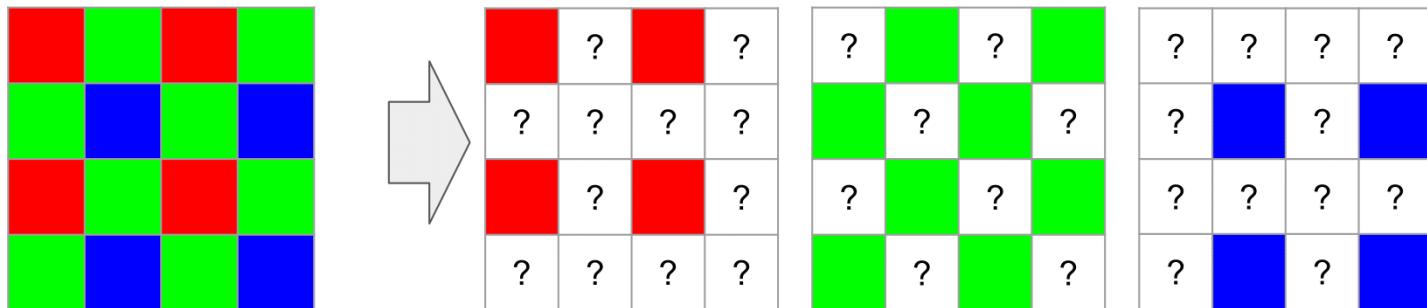
Post-process

Процесс обработки изображений в смартфоне

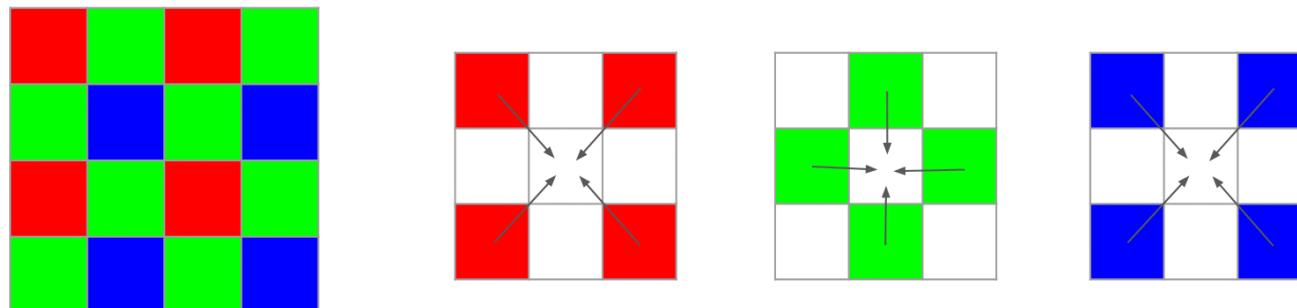
Super Res Zoom

Байеровский шаблон

12 МР сенсор \neq 12 миллионов **RGB** пикселей

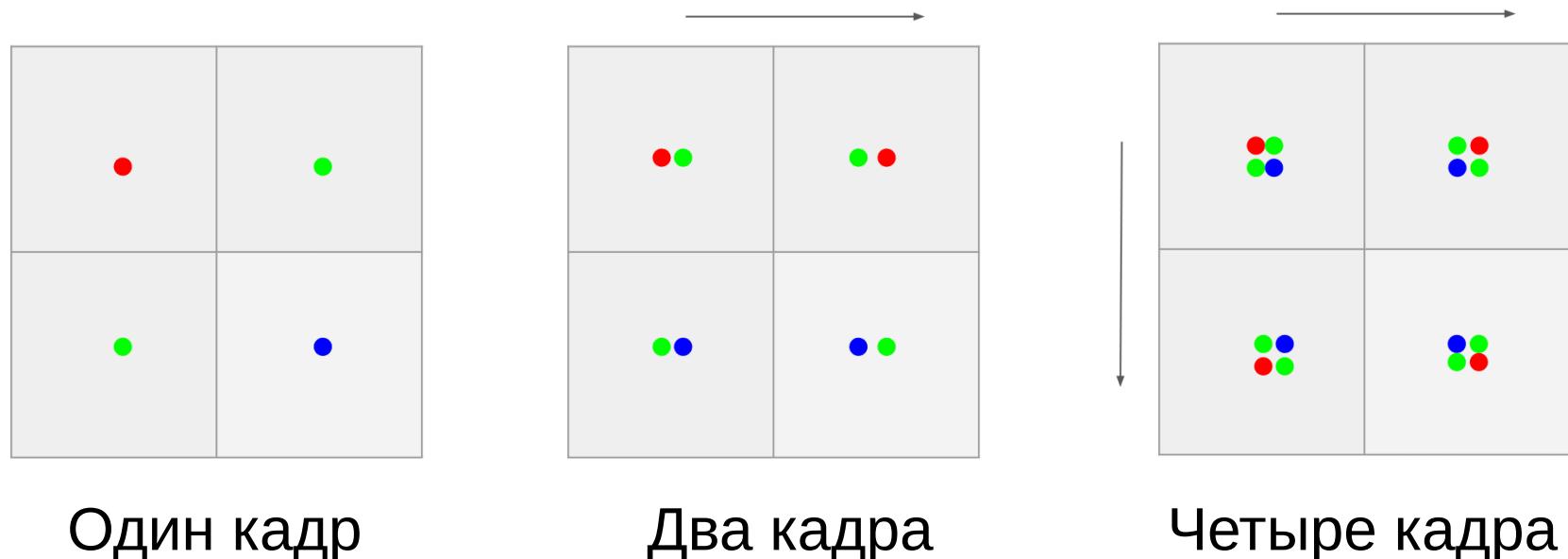


В самом простом случае предположим, что цвет в области постоянный, поэтому заполним недостающие ячейки на основе одноцветных соседей



Беззеркальные камеры

Pixel shift mode (1)



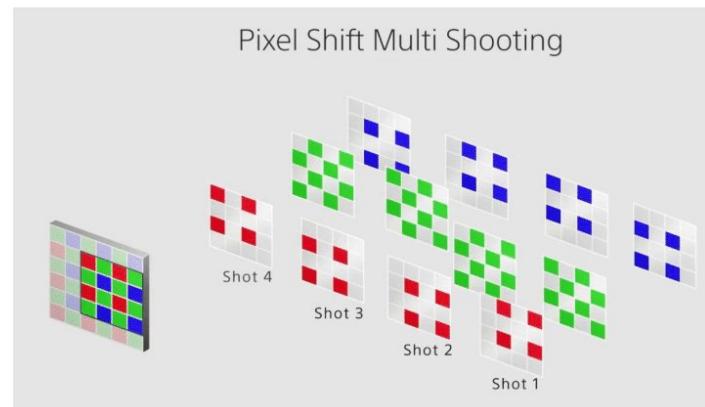
Беззеркальные камеры

Pixel shift mode (2)

Olympus E-M5 Mark II



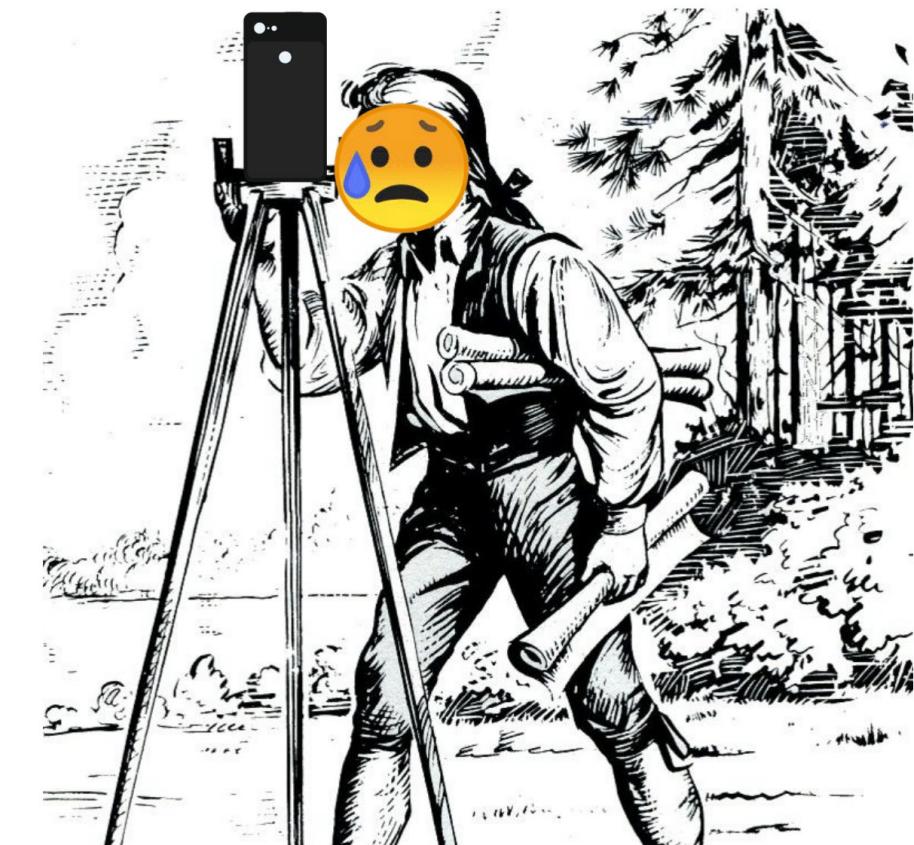
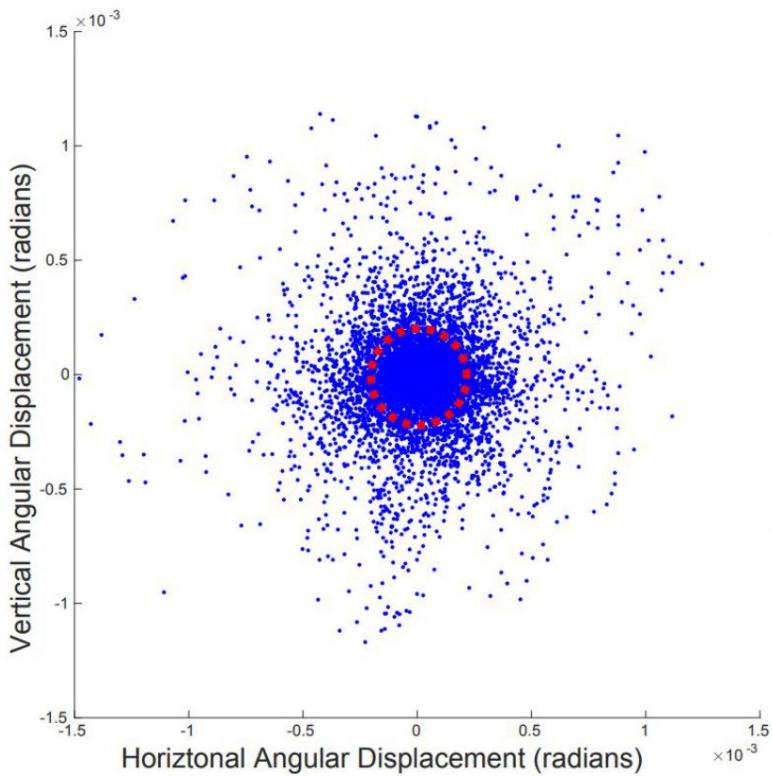
Sony a7R III



Super Res Zoom

Откуда взять дополнительную информацию?

Непроизвольное дрожание рук



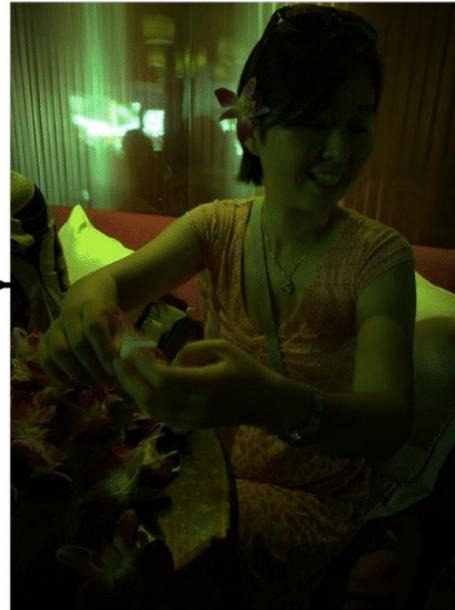
Super Res Zoom

Нужно больше изображений

У смартфона плохая камера, но мощный процессор. Воспользуемся этим! Вместо одной будем снимать сразу несколько фотографий. Такой метод зарекомендовал себя при ночной съёмке



Burst of RAW images



Merged RAW

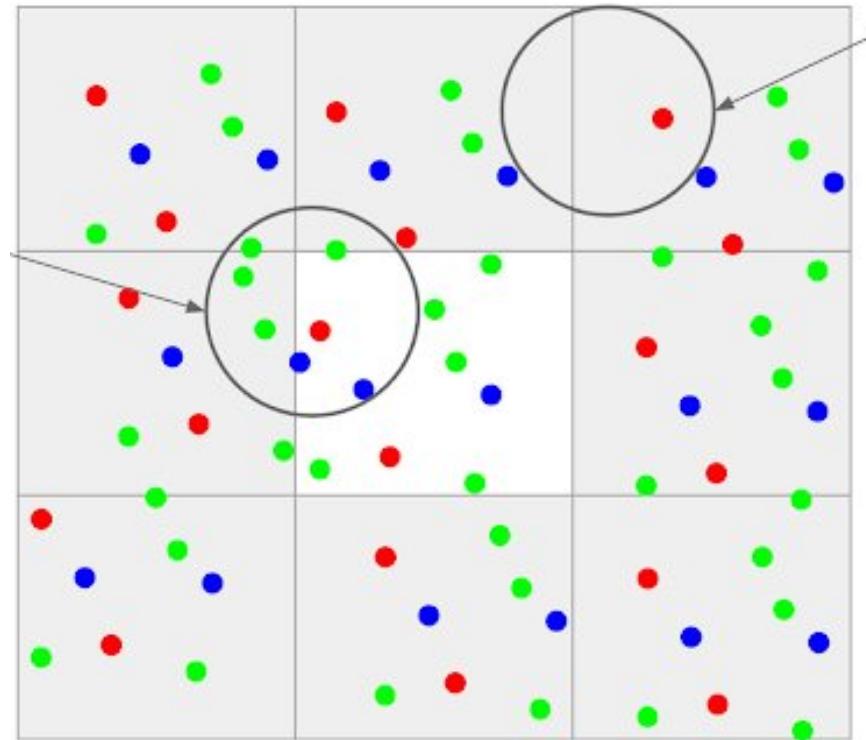


Final Image

Super Res Zoom

Компенсация движения

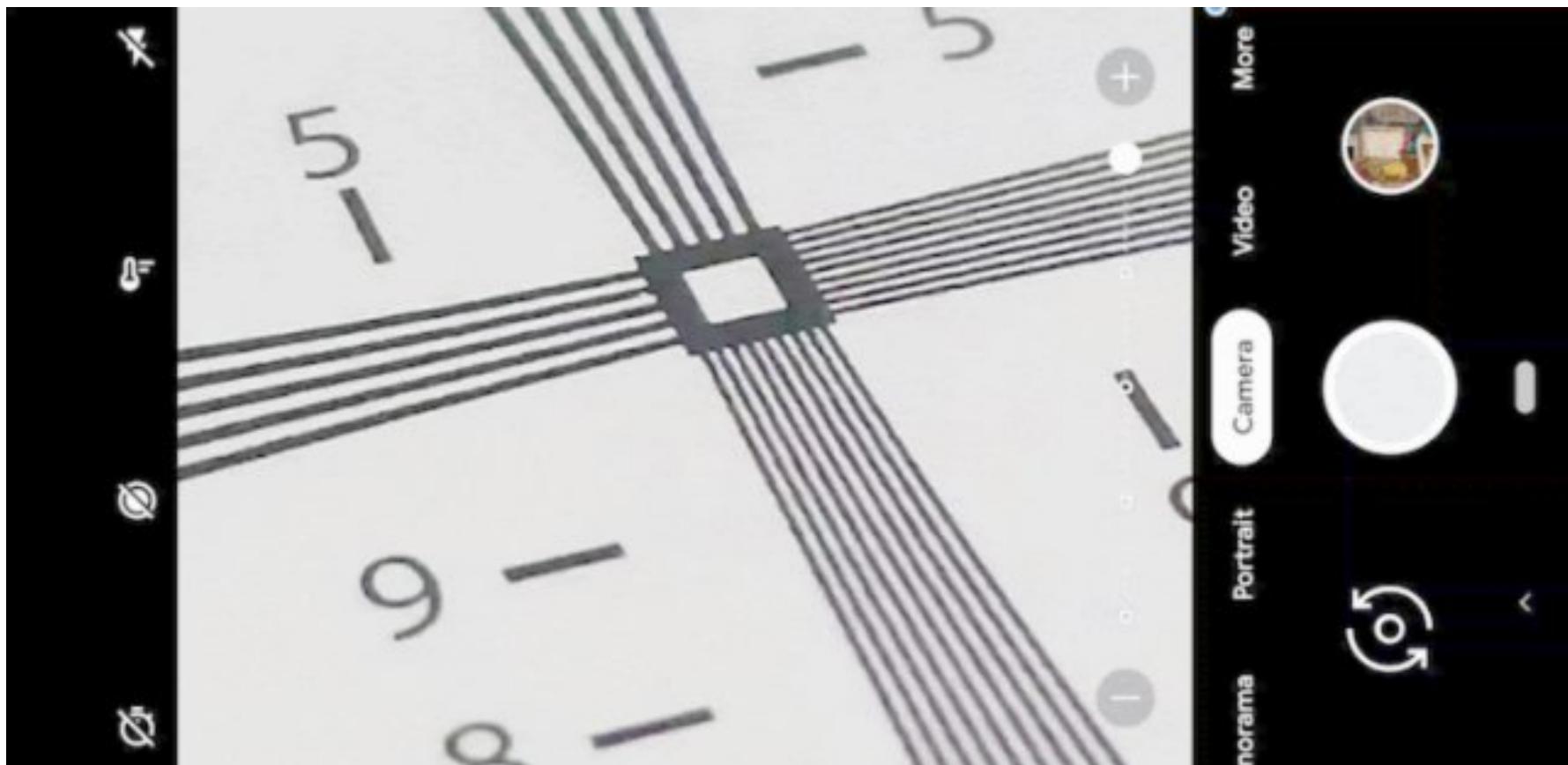
Такой подход даст информацию о цветах в недостающих пикселях



Super Res Zoom

Смартфон на штативе

Модуль оптической стабилизации сам начинает вращать камеру, для получения дополнительной информации



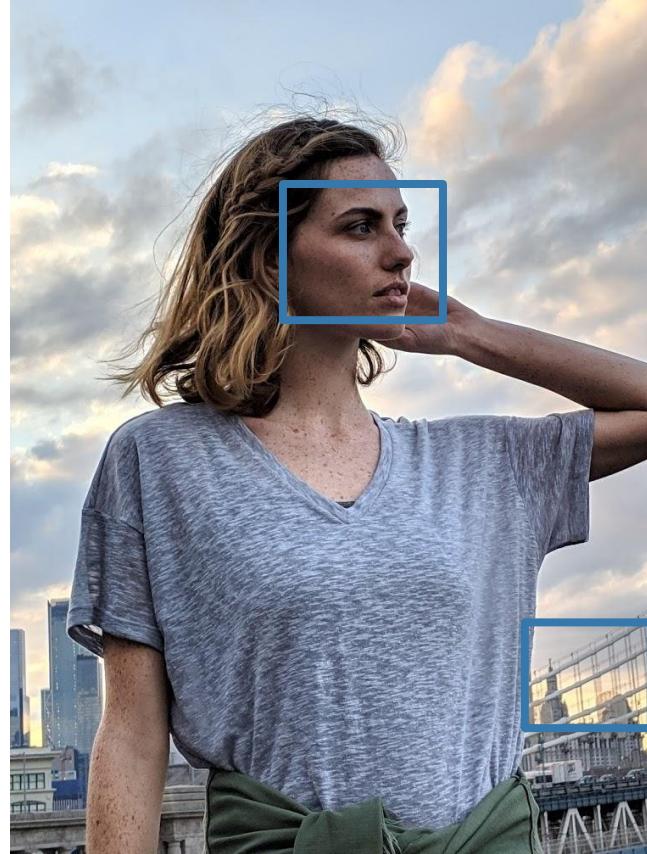
Super Res Zoom

Примеры (1)

Увеличенное изображение снято напрямую со смартфона с помощью зума



Без увеличения



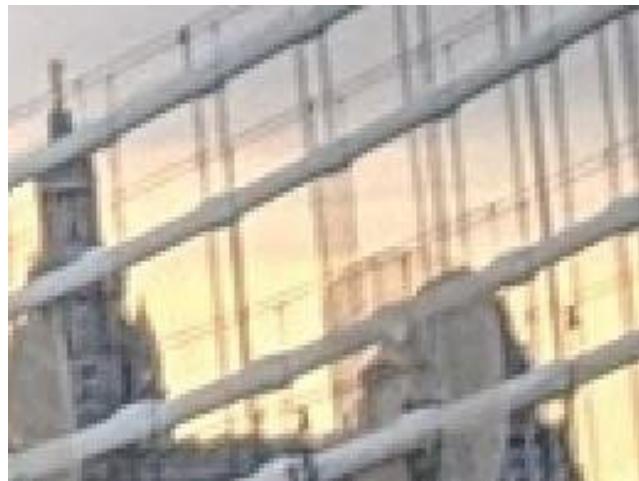
Увеличение 2.65x

Super Res Zoom

Примеры (2)



Без увеличения



Увеличение 2.65x

Выводы

- В последние годы случился заметный скачок в развитии алгоритмов повышения разрешения, особенно это хорошо видно по качеству съёмки смартфонов
- Все это великолепие стало возможно благодаря кардинальному увеличению сложности алгоритмов обработки фотографии

Содержание

- Введение
- Так ёщё мой отец и мой дед делали
- Смартфоны — двигатель прогресса
- **Примеры нейросетевых архитектур**
- Лаборатория и SR

Введение в SR с помощью нейросетей



Традиционные методы интерполяции (например, бикубическая, Ланцоша) имеют ограничения в восстановлении мелких деталей изображения

Даже лучшие классические методы интерполяции не способны полностью восстановить детали при значительном увеличении разрешения



Бикубик



Ланцош

Введение в SR с помощью нейросетей



Нейросетевые методы обучаются на парах изображений низкого и высокого разрешения, что позволяет им восстанавливать текстуры и мелкие детали лучше, чем классические методы



Бикубик

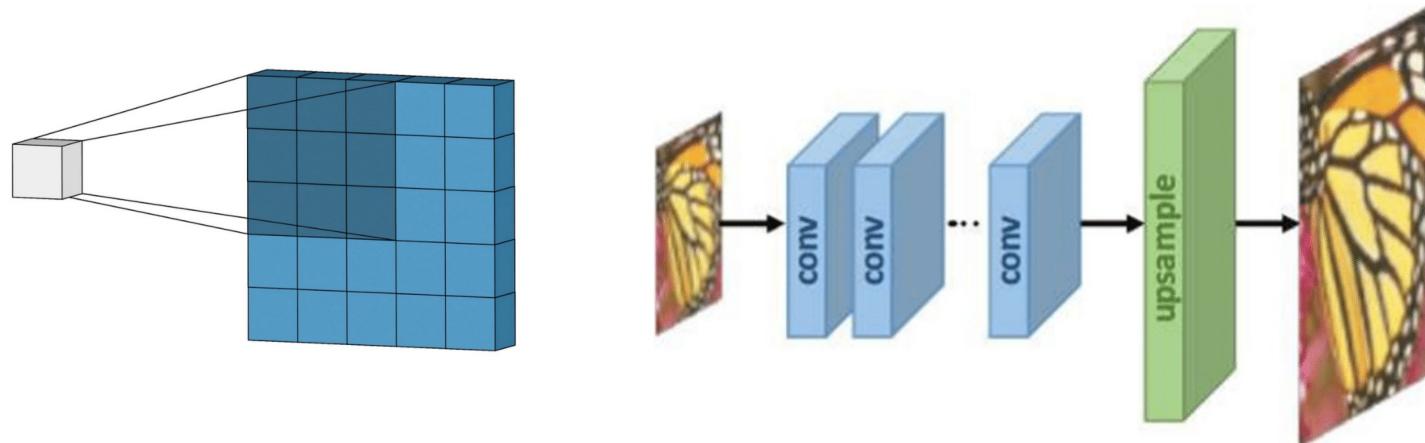


Нейросеть

Свёрточные нейронные сети (CNN) в SR

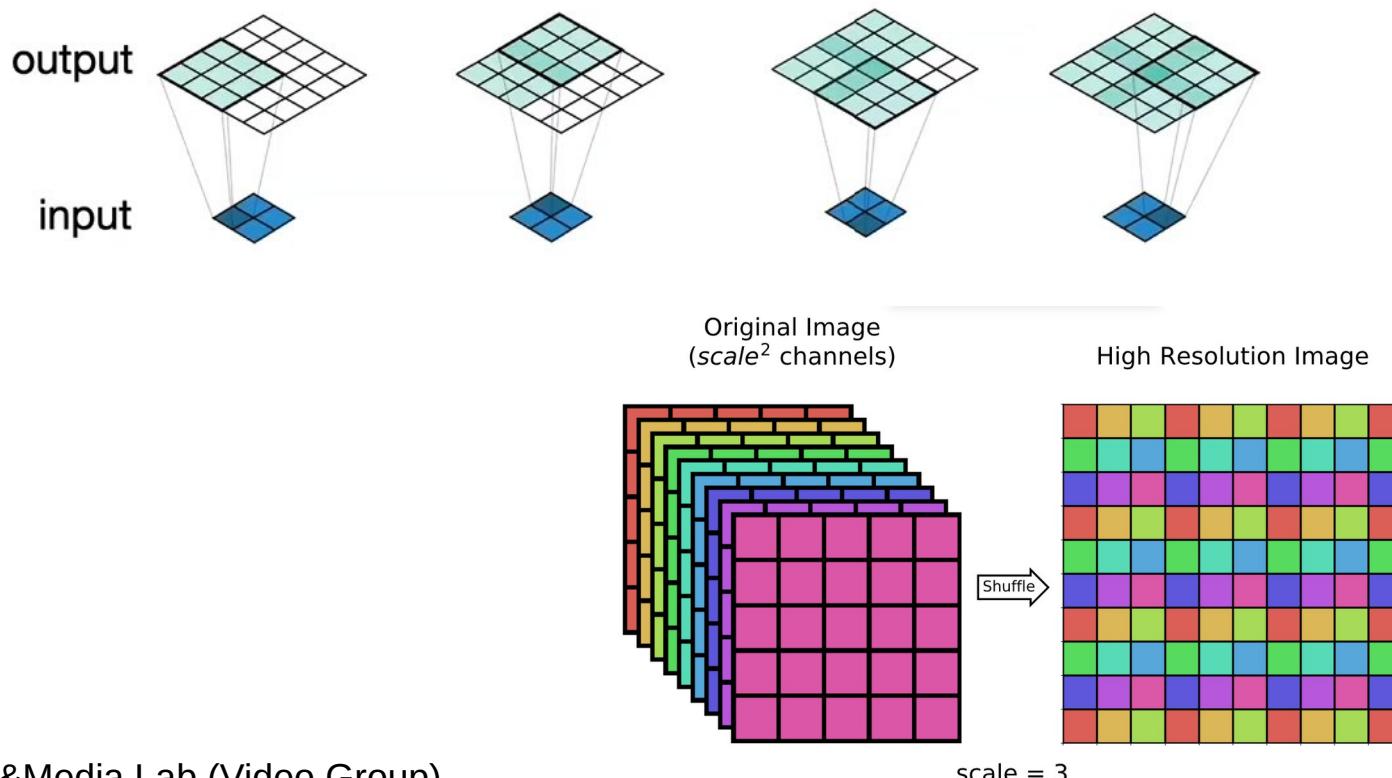
Свёрточные нейронные сети эффективно обрабатывают пространственные данные благодаря свёрткам и нелинейностям

В задачах SR CNN обучаются увеличивать разрешение изображений, используя специальные слои



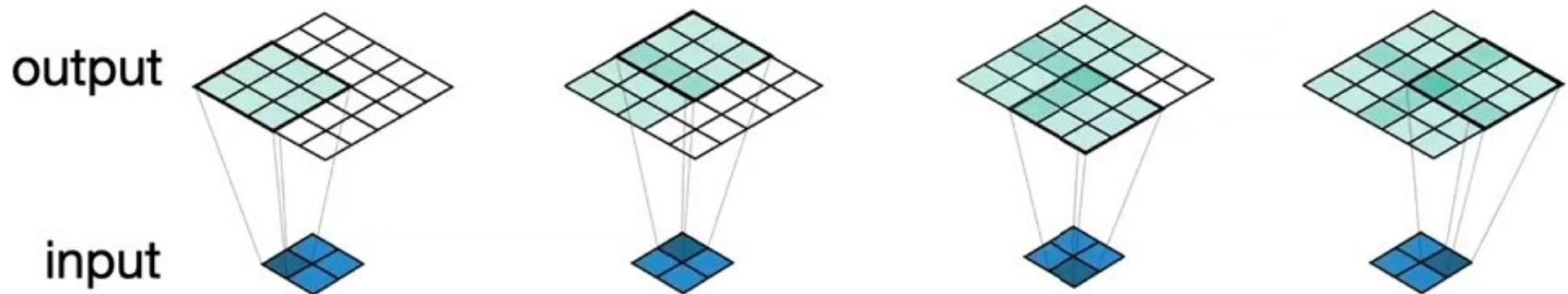
Как CNN увеличивают разрешение изображений

В CNN для SR используются специальные слои для увеличения разрешения, такие как **транспонированные свёртки** или **пиксельная перестановка (Pixel Shuffle)**



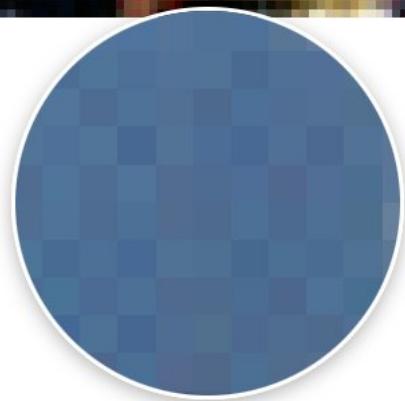
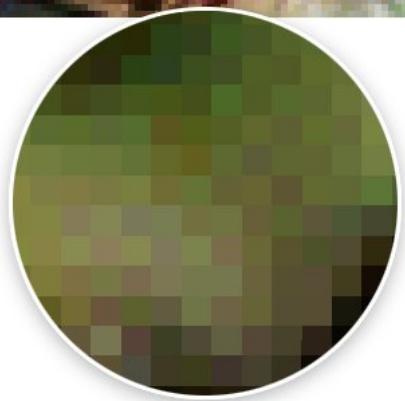
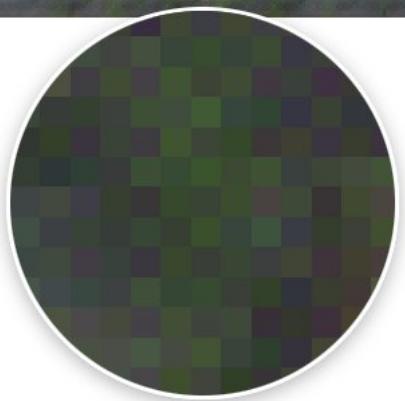
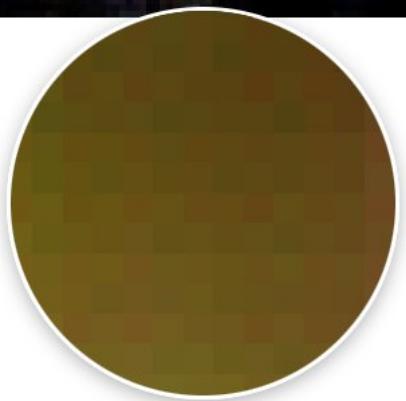
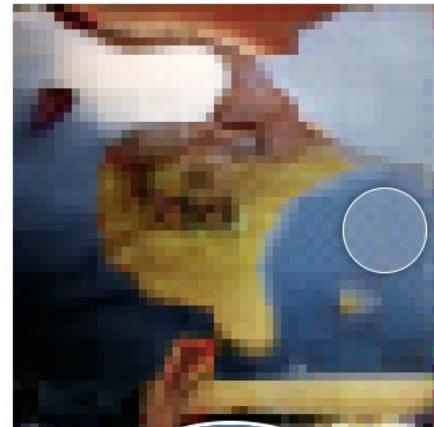
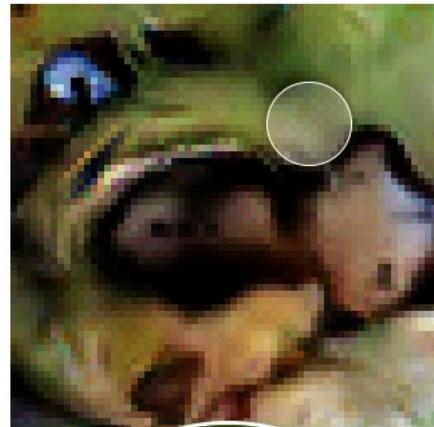
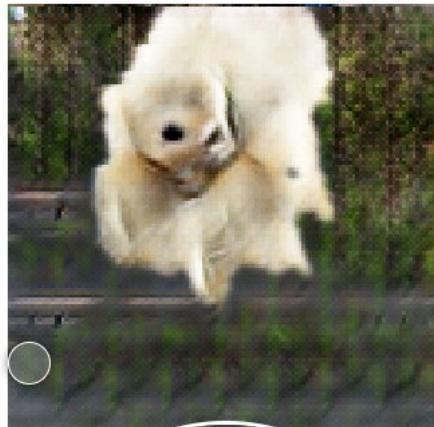
Транспонированные свёртки

Также известны как обратные свёртки или деконволюции
Увеличивают размер карт признаков
Обучаемые фильтры для улучшения качества
Возможны шахматные артефакты



Транспонированные свёртки

Возможны шахматные артефакты, возникающие из-за пересечений обратной свертки

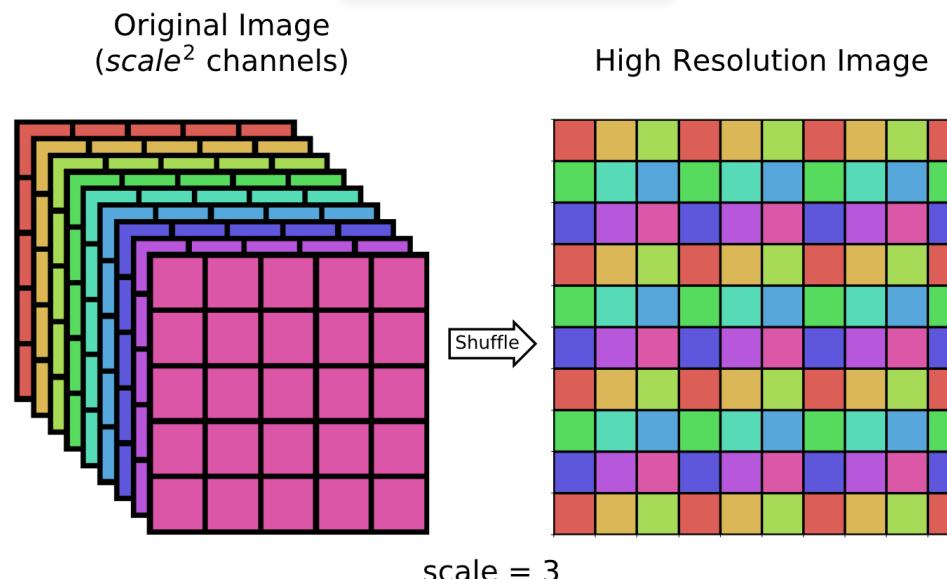


Пиксельная перестановка (Pixel Shuffle)



Переставляет элементы тензора, по сути увеличивая пространственное разрешение

Избегает артефактов, присущих другим методам апсемплинга



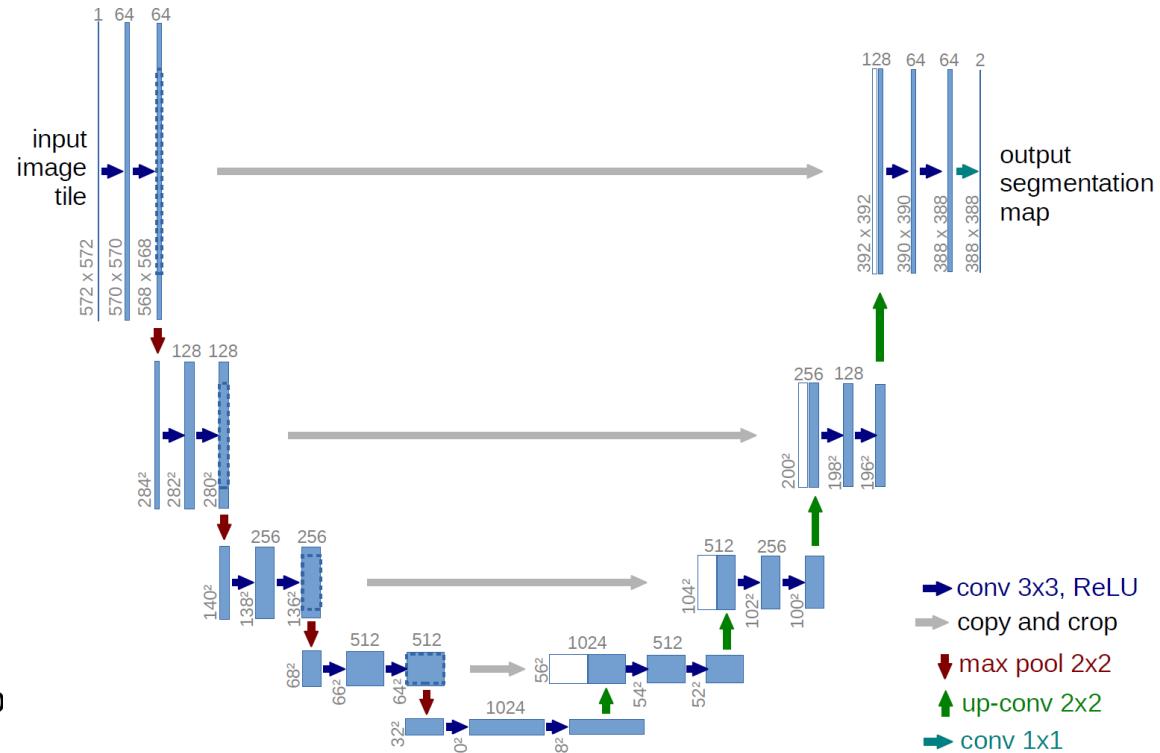
U-Net: применение транспонированных свёрток

Архитектура типа "кодировщик-декодировщик"

Слева обычные свёртки, справа транспонированные

Skip-connections для сохранения детальной информации

Эффективна для задач сегментации и повышения
разрешения



Пример сверточной модели для SR: BasicVSR



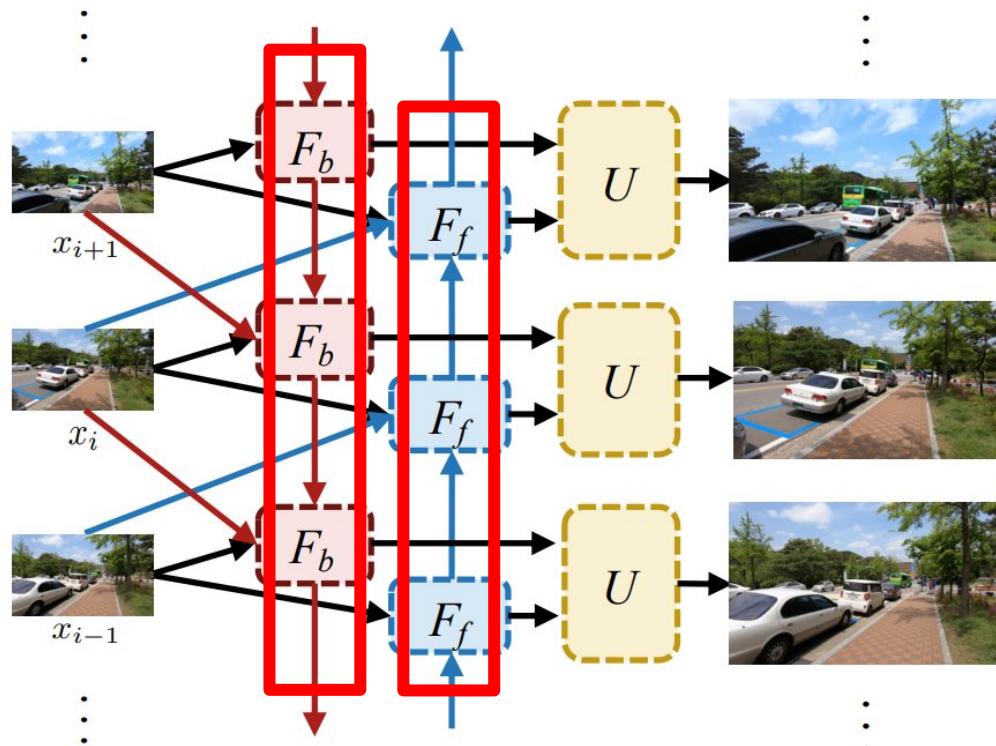
BasicVSR — эффективная модель для повышения разрешения видео на основе CNN

Основные компоненты:

- Двунаправленное распространение информации между кадрами
- **Выравнивание:** использование оптического потока для коррекции движения между кадрами
- **Агрегация:** объединение информации из различных временных шагов
- **Увеличение разрешения:** использование **пиксельной перестановки (Pixel Shuffle)**

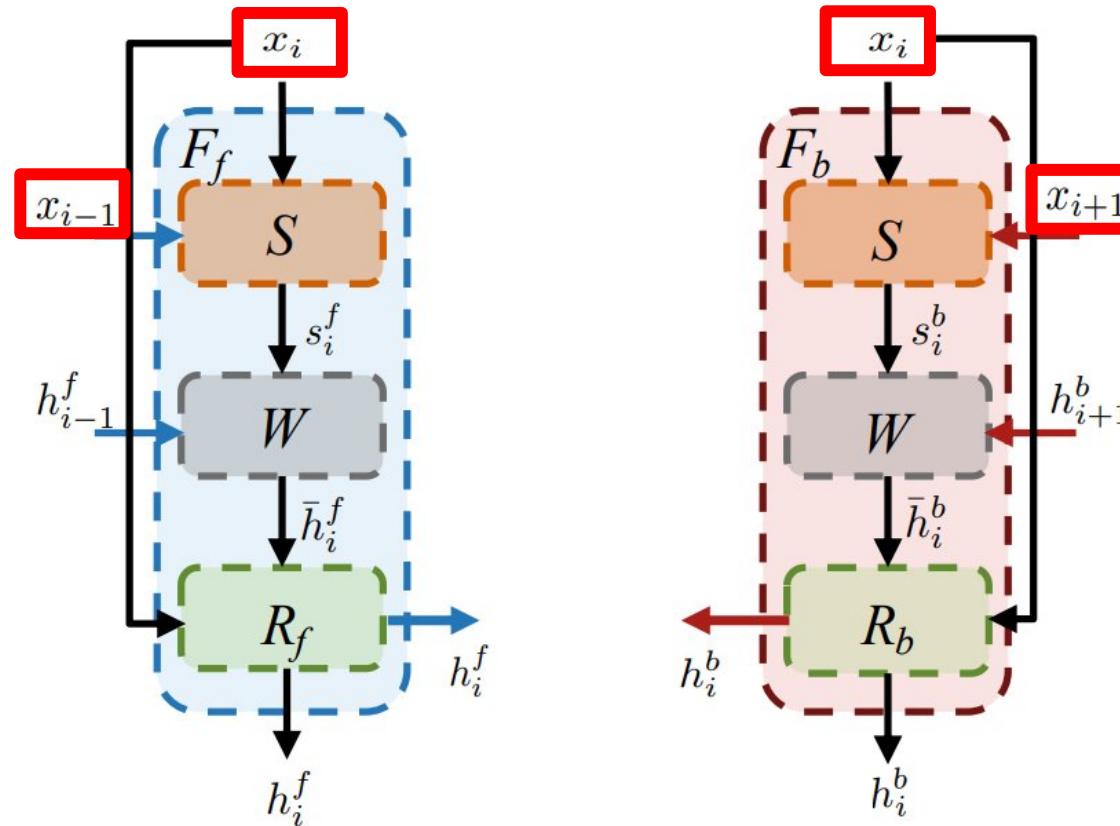
Пример сверточной модели для SR: BasicVSR

Пропагация: двунаправленное распространение информации между кадрами



Пример сверточной модели для SR: BasicVSR

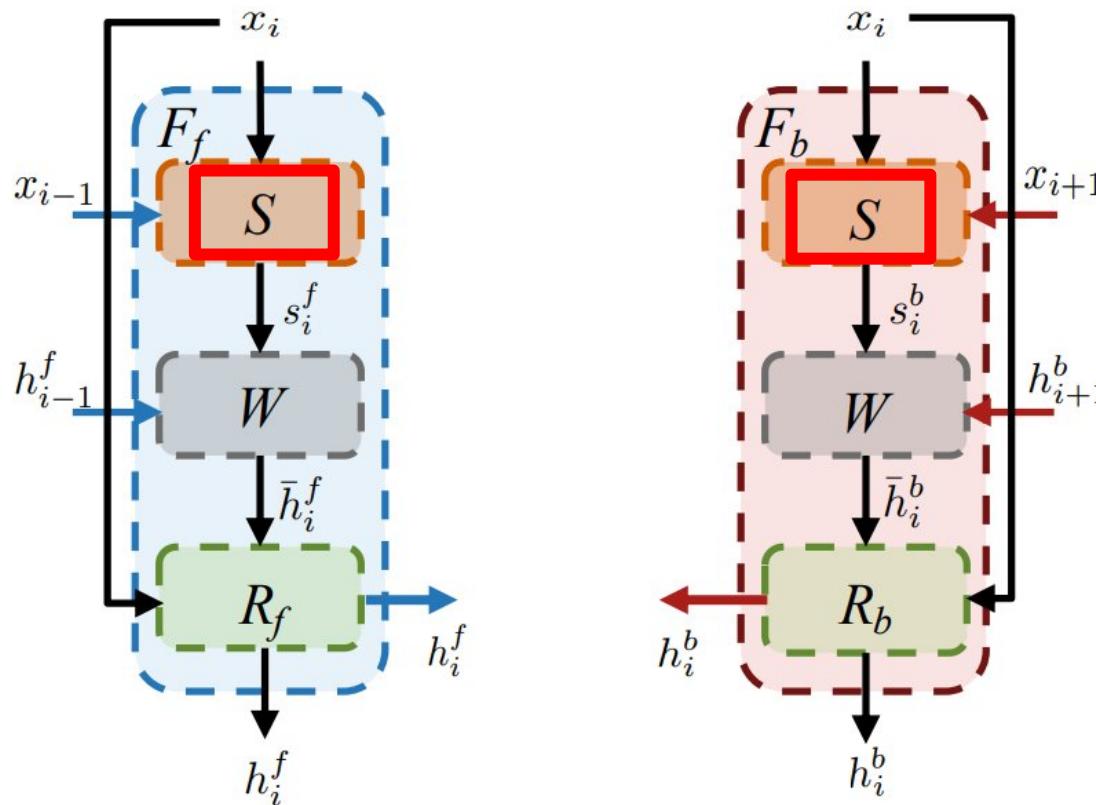
Пропагация: двунаправленное распространение информации между кадрами



Пример сверточной модели для SR: BasicVSR

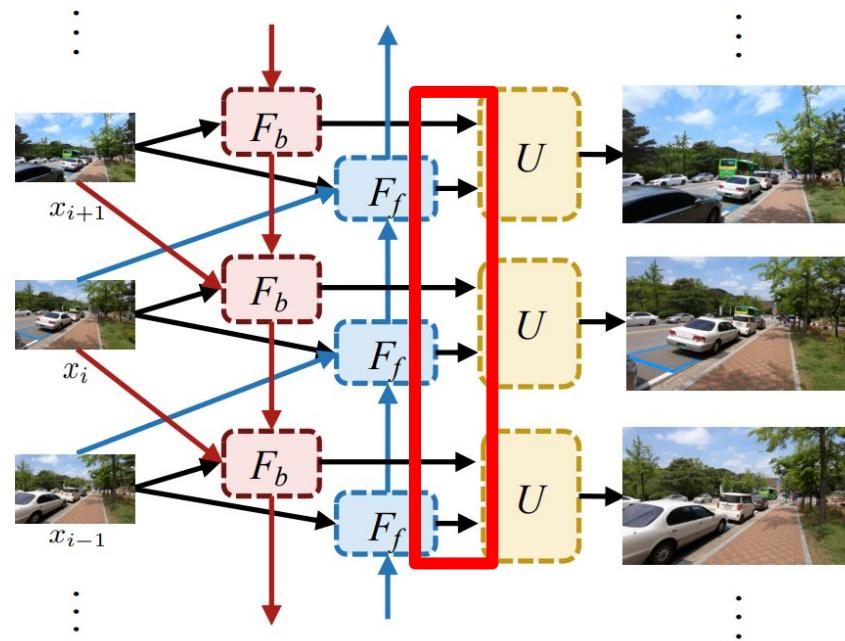


Выравнивание: использование оптического потока для коррекции движения между кадрами



Пример сверточной модели для SR: BasicVSR

Агрегация: объединение информации из различных временных шагов

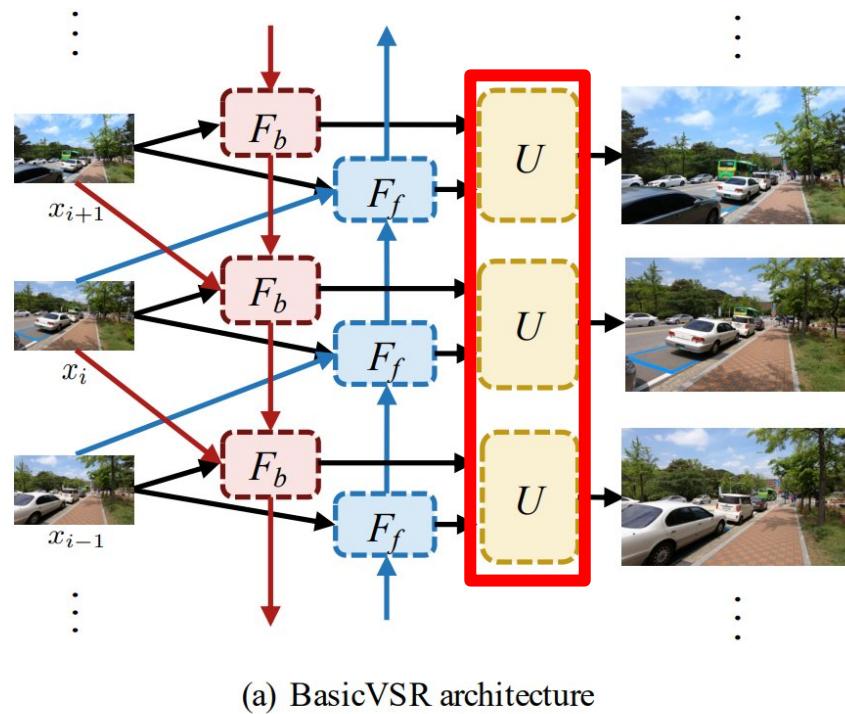


(a) BasicVSR architecture

Пример сверточной модели для SR: BasicVSR



Увеличение разрешения: использование пиксельной перестановки (Pixel Shuffle)



Архитектура BasicVSR и её работа



Двунаправленная рекуррентная сеть передаёт информацию из прошлых и будущих кадров для улучшения текущего

Оптический поток используется для точного выравнивания признаков между кадрами

Простая и эффективная структура позволяет достигать высоких результатов при меньшей сложности модели

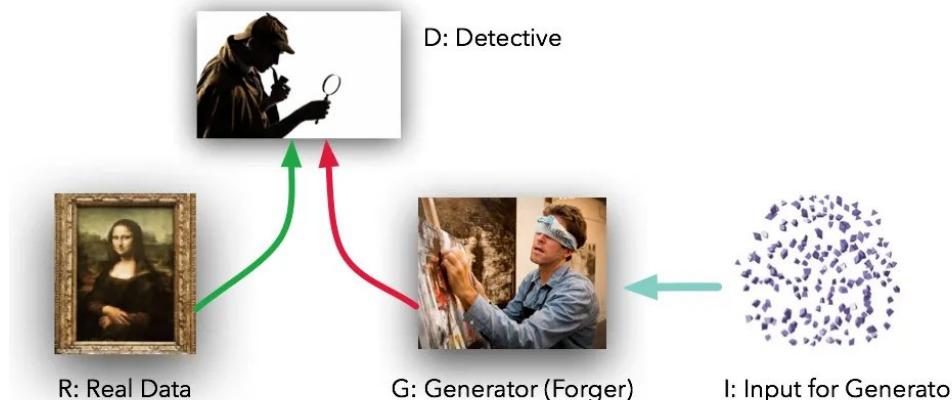
Генеративно-состязательные сети (GAN) в SR



GAN состоят из двух сетей:

- **Генератор (G)**: генерирует новые данные из шума или низкого разрешения
- **Дискриминатор (D)**: пытается отличить реальные данные от сгенерированных

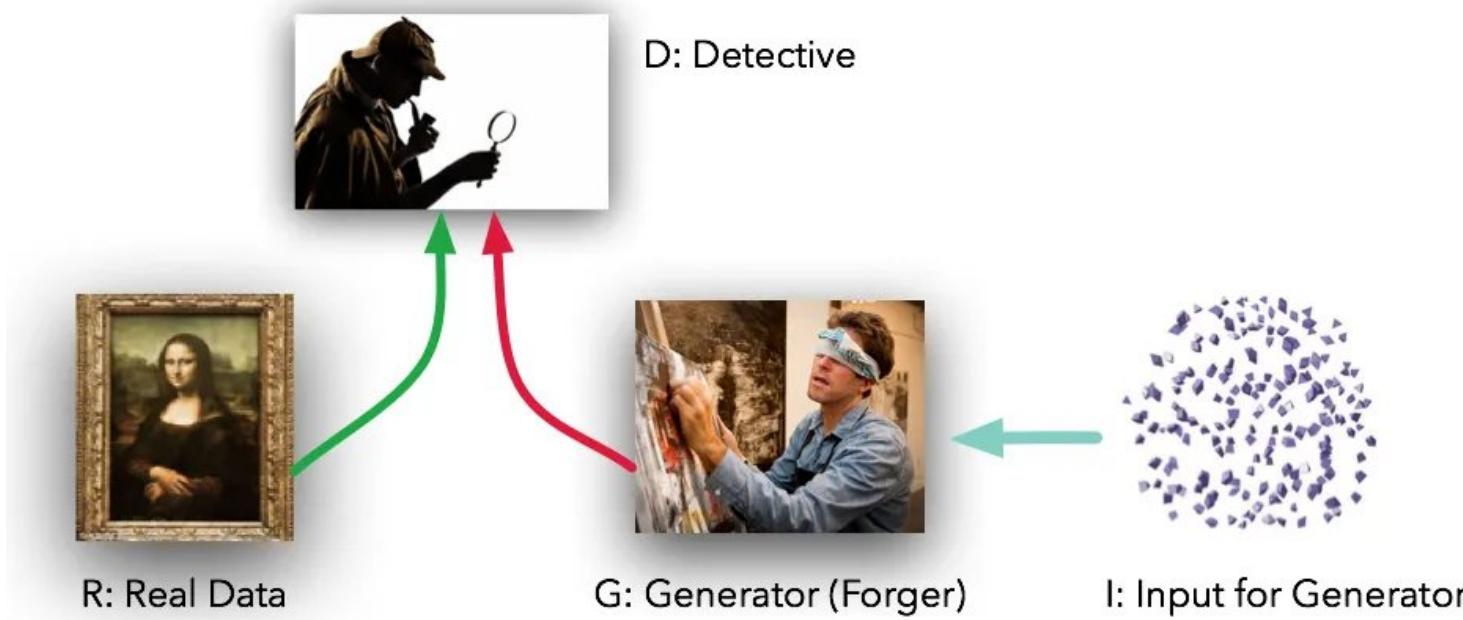
Обучение происходит через соревнование между G и D, что улучшает качество генерации



Генеративно-состязательные сети (GAN) в SR



Преимущество в SR: способность генерировать реалистичные текстуры и детали, недоступные классическим методам

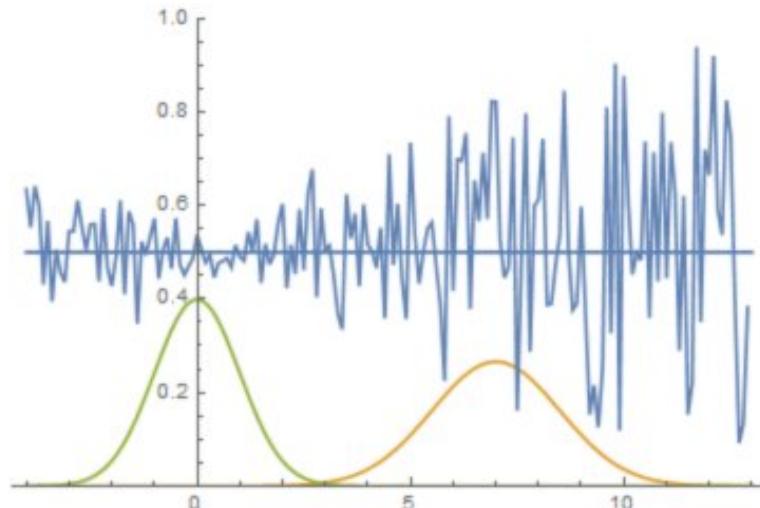


Основы генеративно-состязательных сетей (GAN)

Генератор (G): преобразует входные данные в реалистичные изображения

Дискриминатор (D): оценивает вероятность того, что изображение является реальным

Цель: G учится обманывать D, в то время как D учится лучше распознавать сгенерированные изображения



Неподвижная гауссиана — плотность распределения реальных данных, движущаяся гауссиана — плотность распределения генерируемых примеров, синяя кривая — результат работы дискриминатора, т.е. вероятность примера быть настоящим

VideoGigaGAN: детализированное видео SR с помощью GAN



VideoGigaGAN расширяет возможности GAN для обработки видео

Особенности модели:

- Использует масштабную архитектуру GAN, обученную на больших объемах данных
- **Временные слои (Temporal Layers)**: интегрированы для обеспечения согласованности между кадрами
- Применяет **антиалиасинг** в кодере для уменьшения мерцаний и артефактов

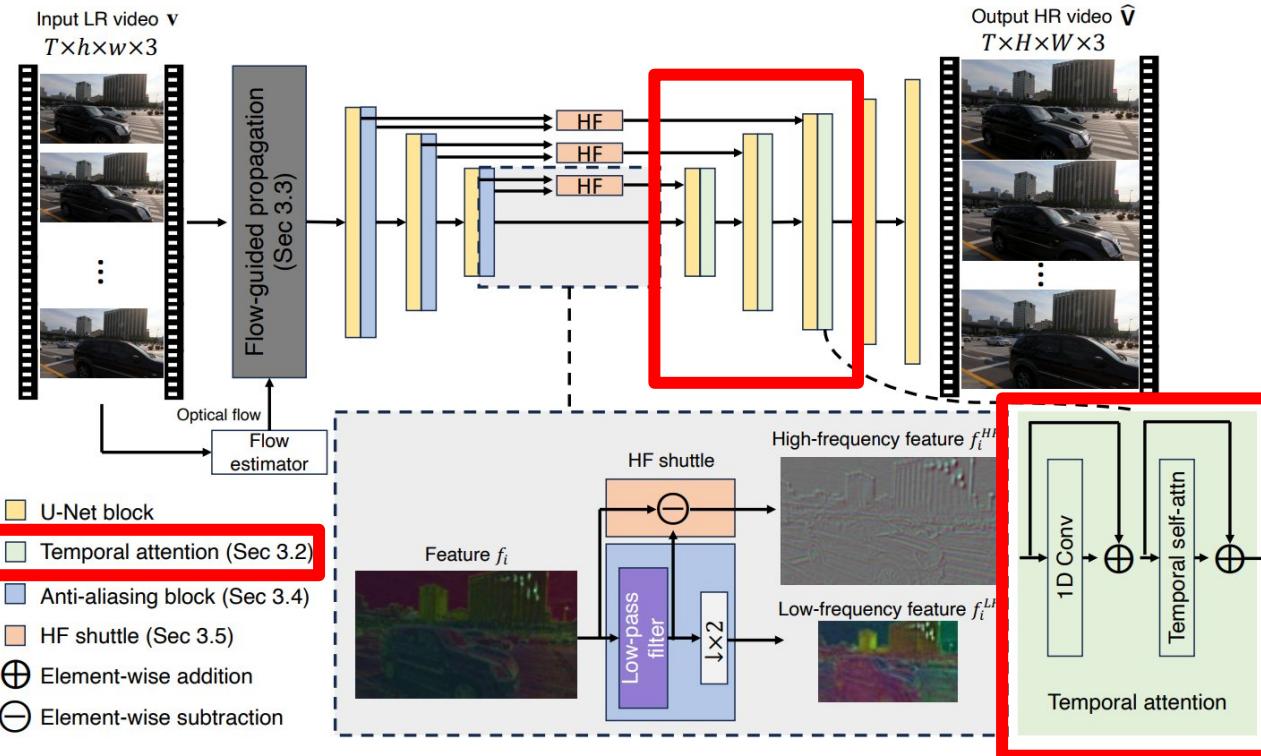
Интересно с точки зрения SR: Способна восстанавливать мелкие детали и текстуры в видео, обеспечивая высокое качество визуализации

Как VideoGigaGAN обеспечивает временную согласованность



Временные слои:

- Анализируют группы по 3 соседних кадра для отслеживания движения и изменений
- Сравнивают и согласовывают детали между кадрами для создания плавного видео

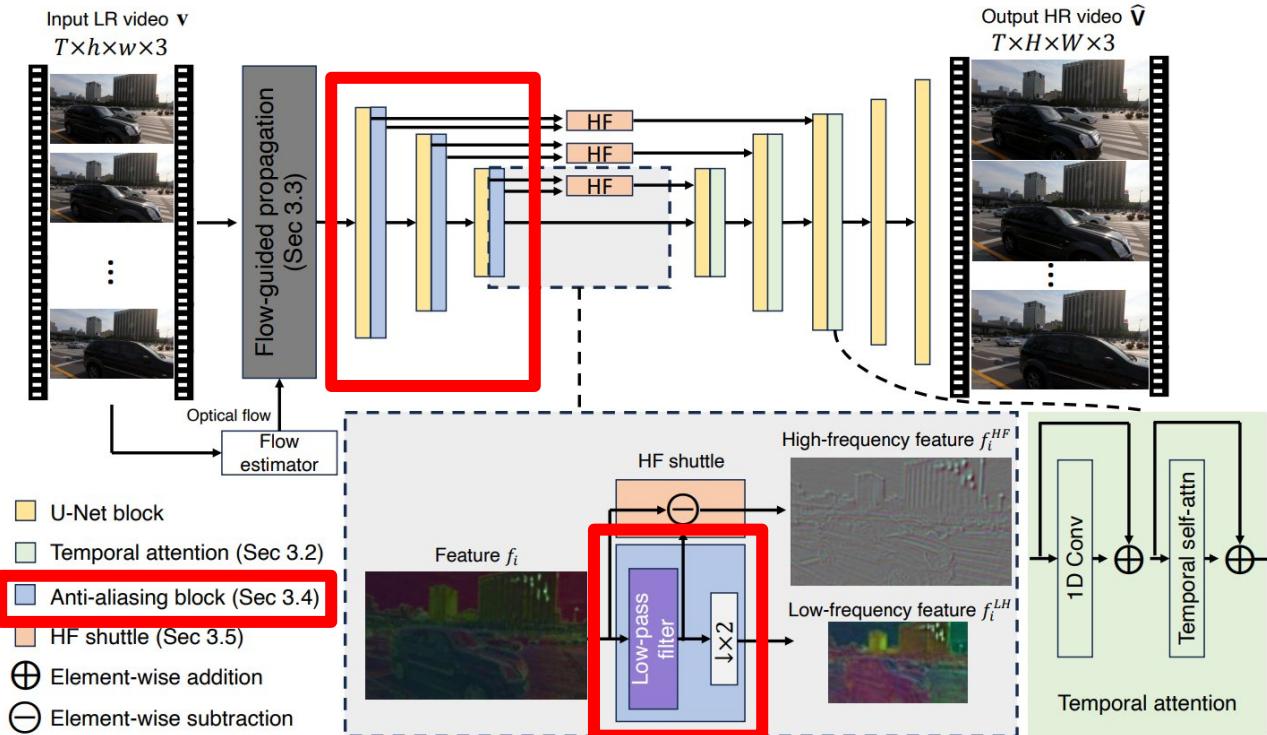


Как VideoGigaGAN обеспечивает временную согласованность



Антиалиасинг в кодере:

- Применение фильтров низких частот при понижении размерности
- Подавляет артефакты и мерцания, связанные с понижением дискретизации

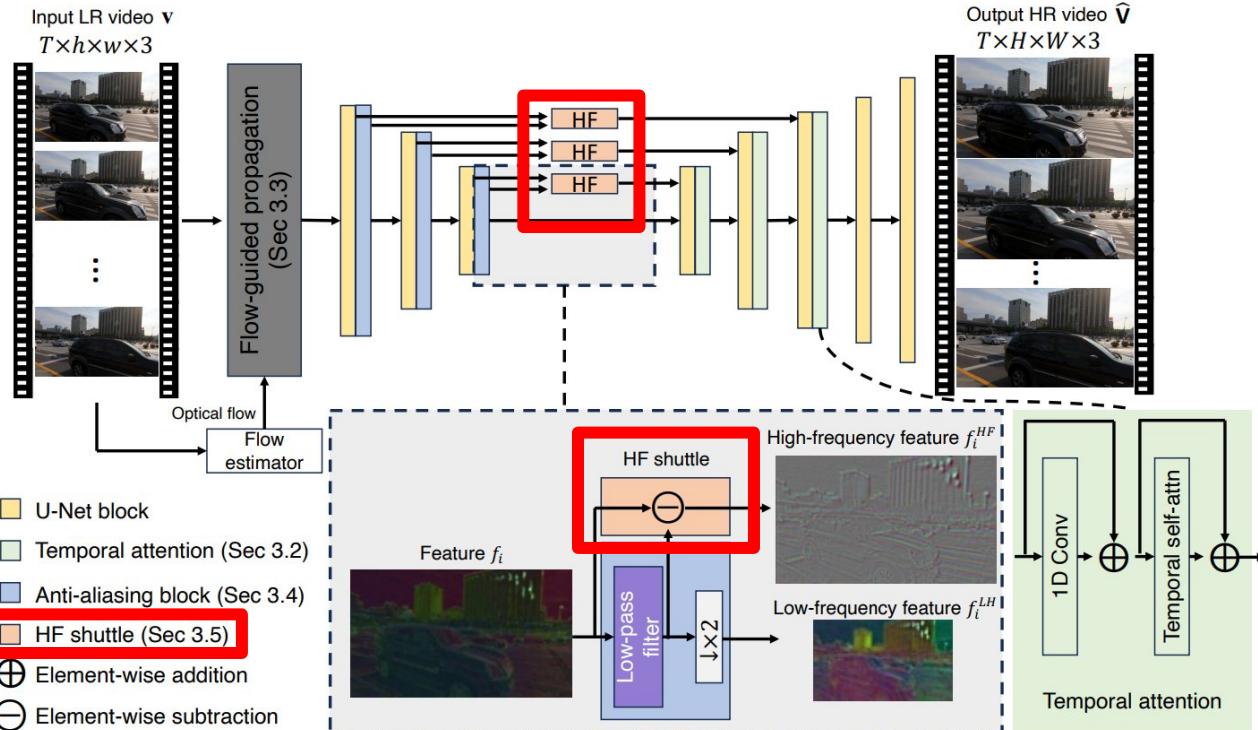


Как VideoGigaGAN обеспечивает временную согласованность

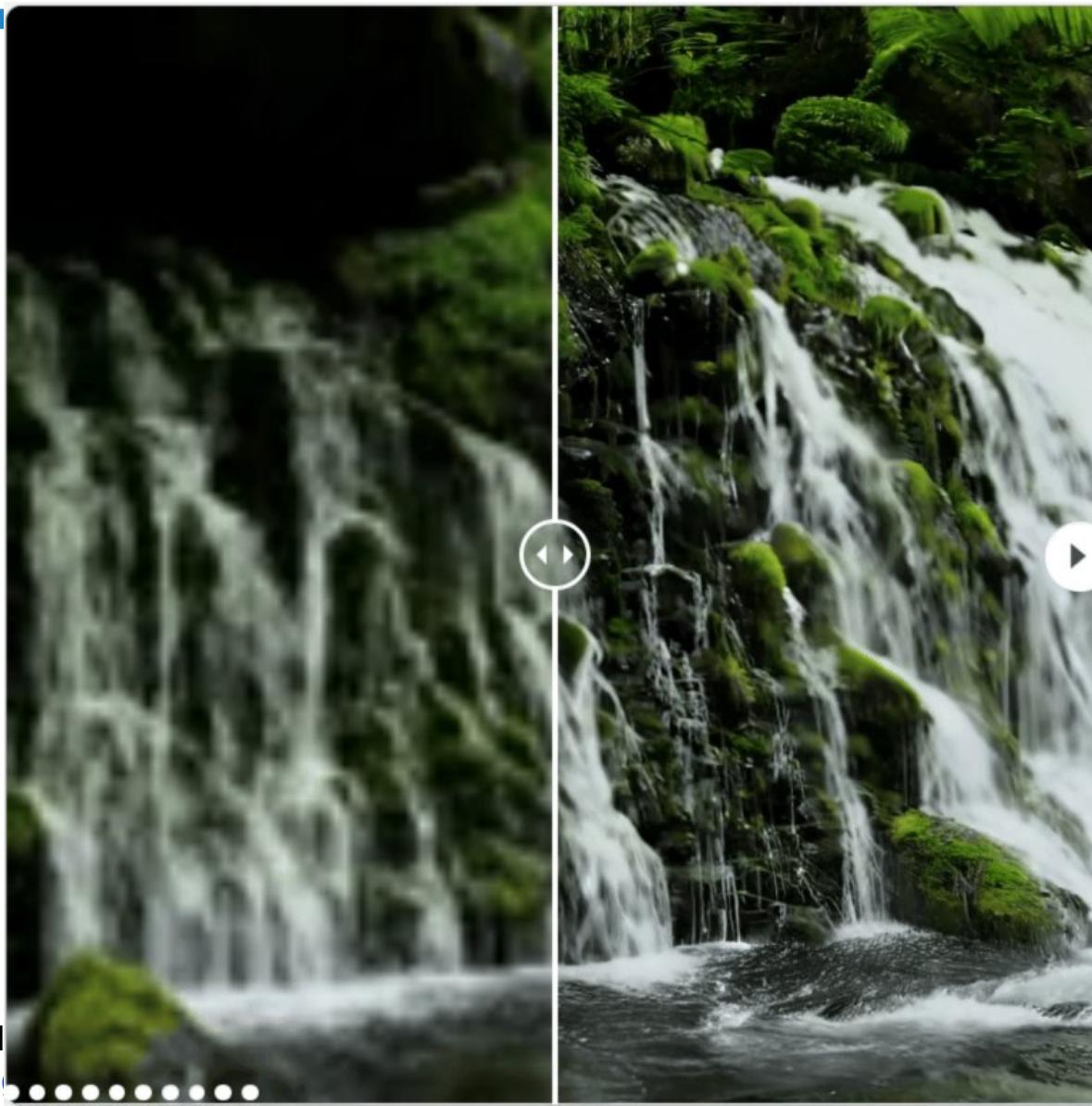


Механизм высокочастотного шаттла (HF shuttle):

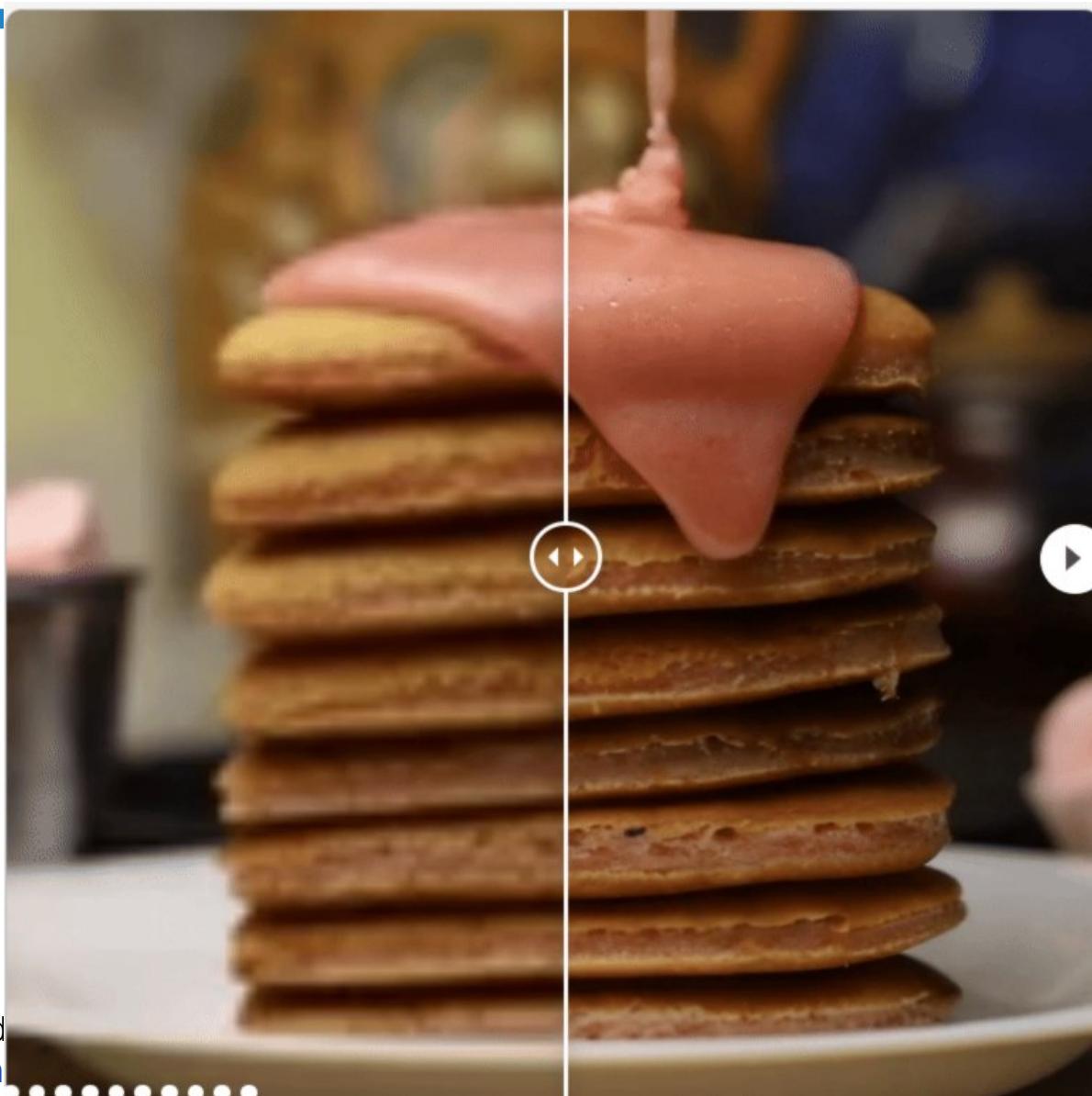
- Передаёт высокочастотные признаки напрямую к декодеру.
- Улучшает детализацию в восстановленных кадрах



Примеры работы VideoGigaGAN



Примеры работы VideoGigaGAN



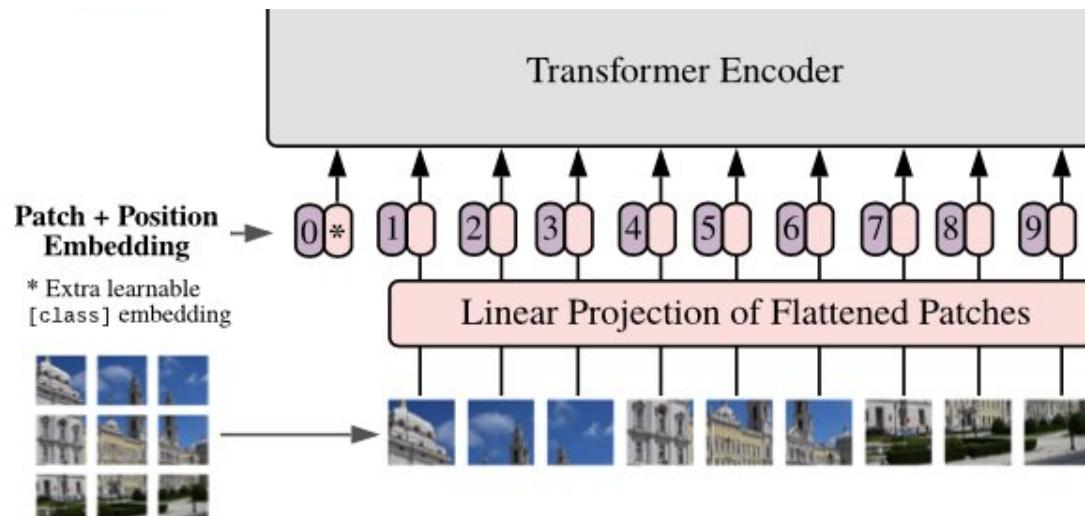
Трансформеры в компьютерном зрении



Трансформеры используют **механизм внимания** (**Attention**) для обработки последовательных данных

Позволяют эффективно моделировать **дальние зависимости** в данных

Изображение и видео: адаптированы через разбиение на патчи и обработку их как последовательностей



Механизм внимания (Attention) в трансформерах



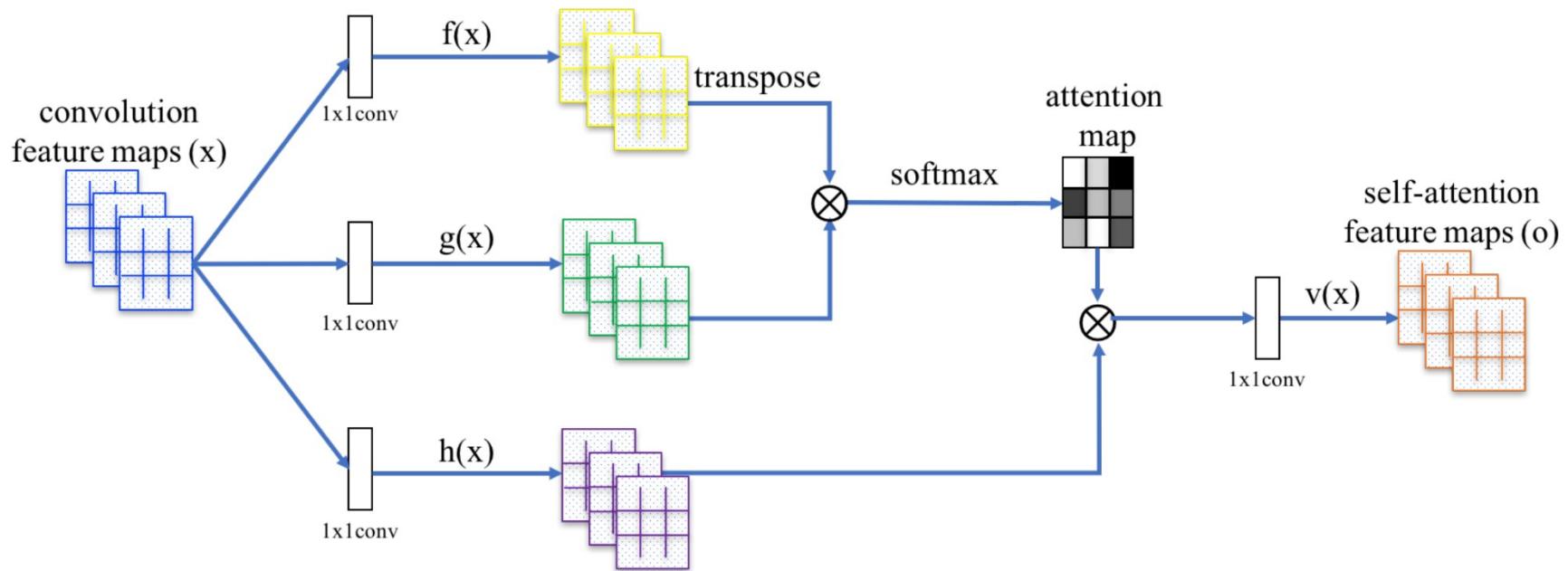
Внимание позволяет модели фокусироваться на важных частях входных данных

Самовнимание (Self-Attention): внутри одной последовательности (например, внутри кадра)

Взаимное внимание (Cross-Attention): между разными последовательностями (например, между кадрами)

Применение в видео: моделирование взаимодействий между кадрами для улучшения качества восстановления

Механизм внимания (Attention) в трансформерах



VRT: Video Restoration Transformer

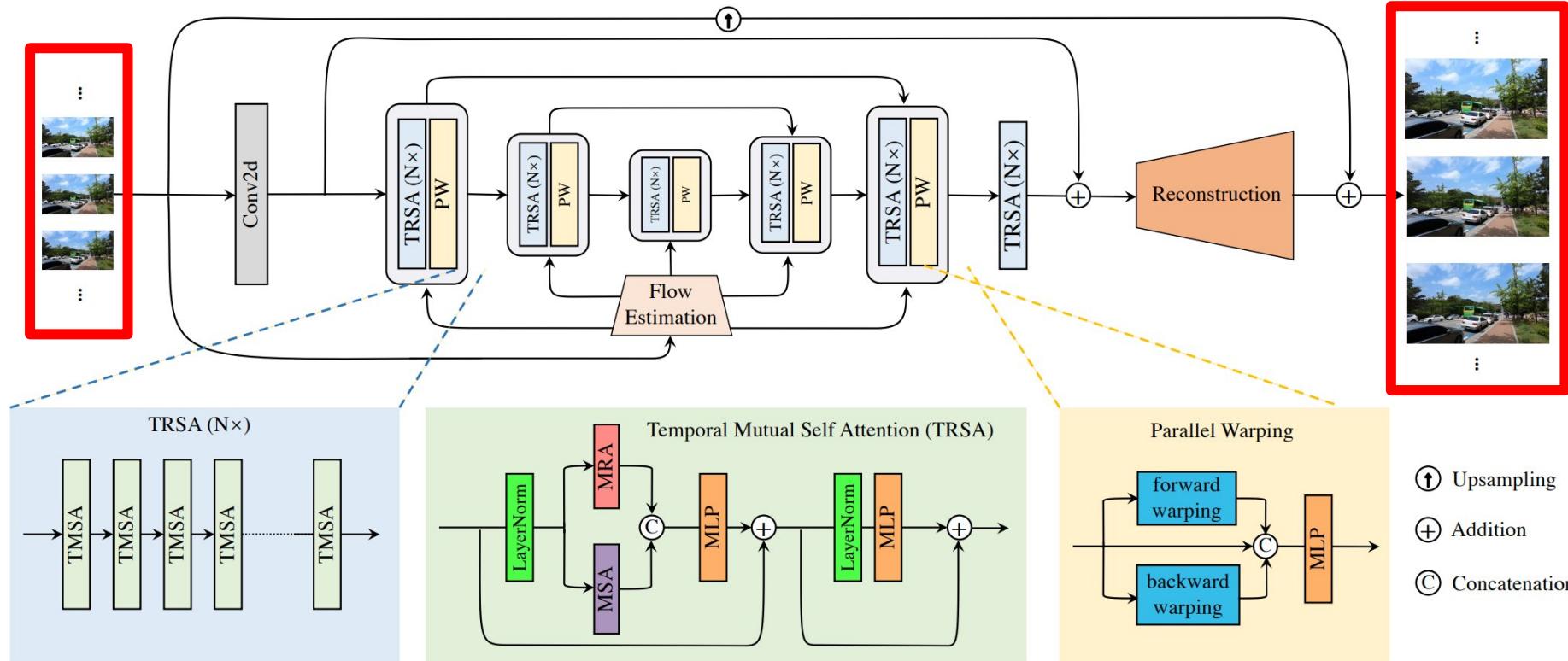
VRT применяет трансформеры для восстановления видео с учётом временных зависимостей

Особенности модели:

- **Параллельная обработка:** все кадры обрабатываются одновременно, что ускоряет вычисления
- **Темпоральное взаимное самовнимание (TMSA):** объединяет информацию между кадрами для лучшего выравнивания и восстановления

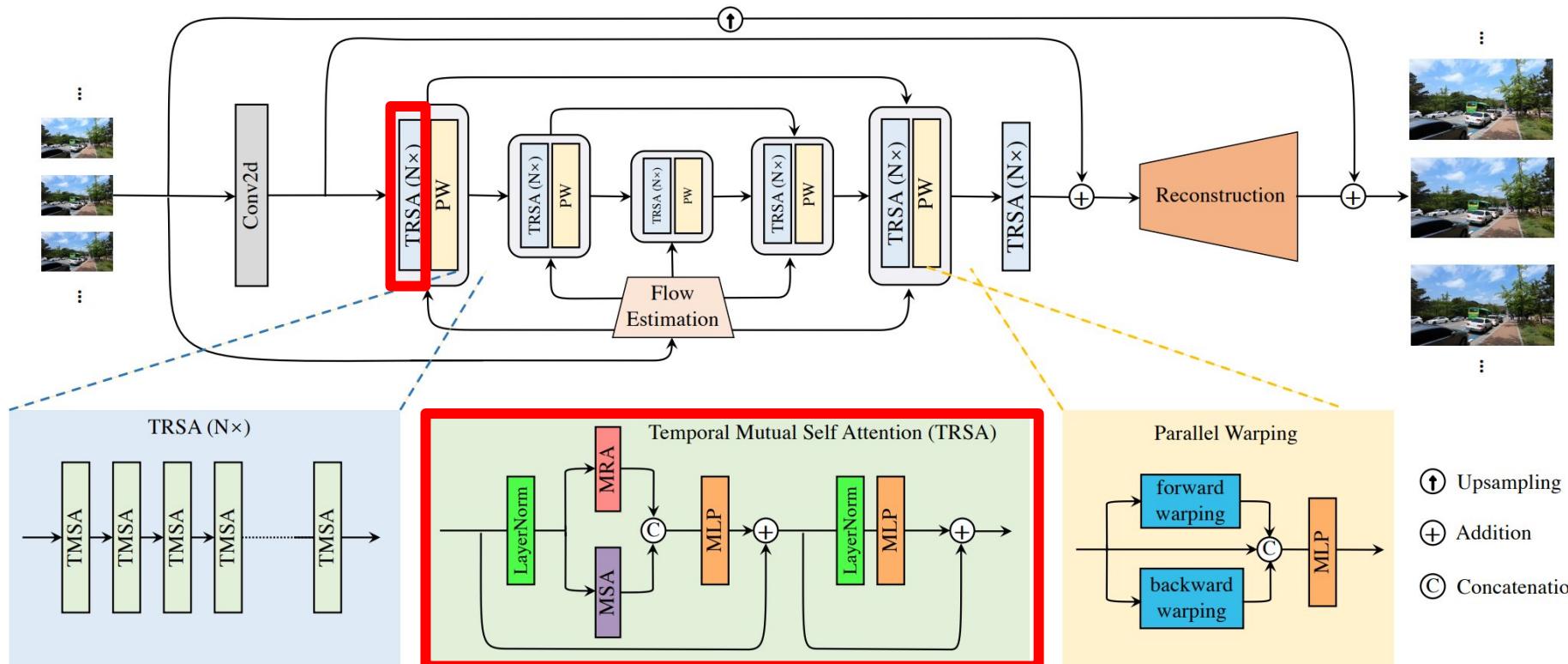
VRT: Video Restoration Transformer

Параллельная обработка: все кадры обрабатываются одновременно, что ускоряет вычисления



VRT: Video Restoration Transformer

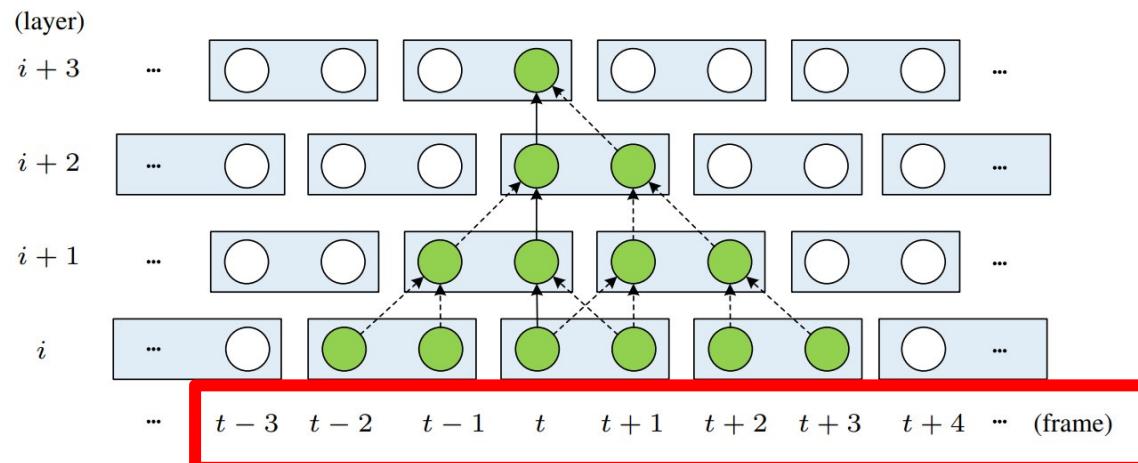
Темпоральное взаимное самовнимание (TMSA):
 объединяет информацию между кадрами для лучшего
 выравнивания и восстановления



Как VRT моделирует временные зависимости

Темпоральное взаимное самовнимание (TMSA):

- Делит видео на небольшие клипы для обработки
- **Взаимное внимание** между кадрами для выравнивания и объединения информации
- **Самовнимание** внутри кадров для улучшения локальных признаков



Диффузионные модели в компьютерном зрении



Диффузионные модели обучаются превращать шум в данные путём обратного диффузионного процесса

Применяются для генерации изображений, заполнения пропусков и других задач

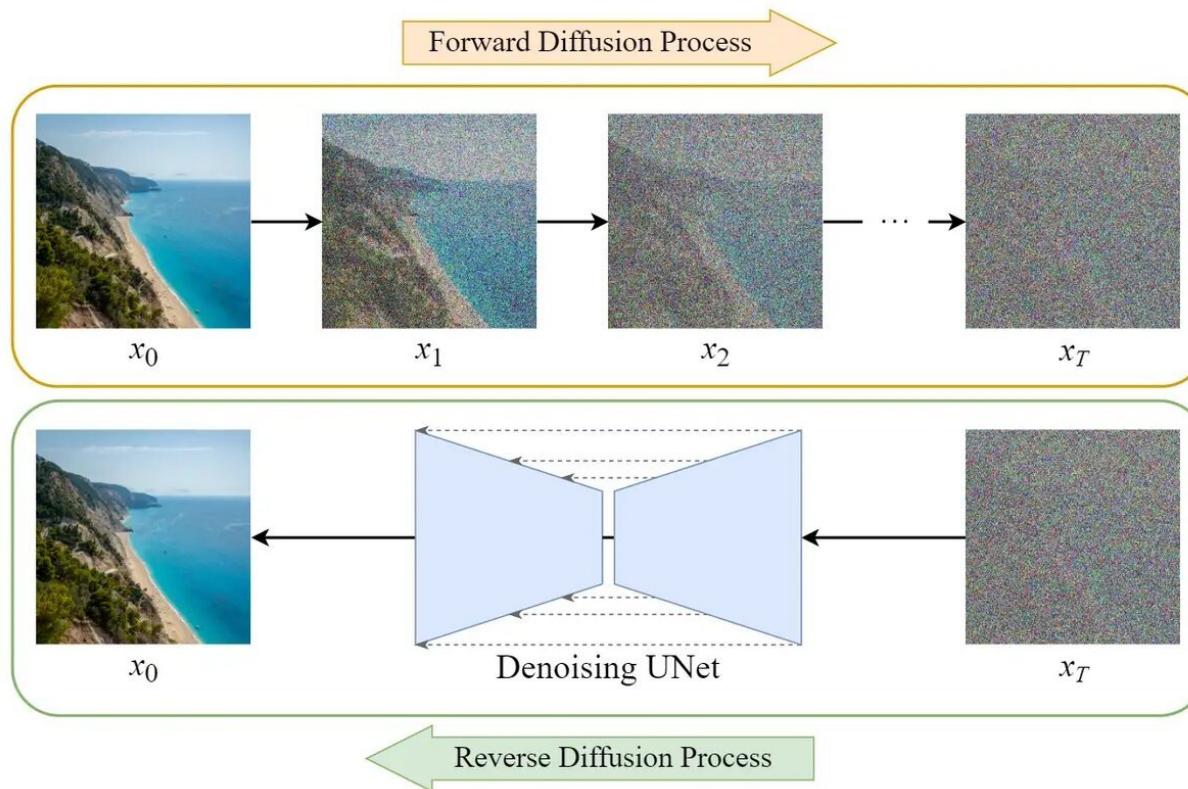
Предоставляют высокое качество генерации с реалистичными деталями

Недостатки:

- Требуют больших вычислительных ресурсов
- Долгое время генерации изображений

Диффузионные модели в компьютерном зрении

Диффузионные модели обучаются превращать шум в данные путём обратного диффузионного процесса



Основы диффузионных моделей

Прямой процесс: постепенное добавление шума к данным

Обратный процесс: модель учится удалять шум, восстанавливая данные

Преимущества:

- Способность моделировать сложные распределения данных
- Высокое качество сгенерированных результатов

Upscale-A-Video: применение диффузионных моделей в SR



Upscale-A-Video использует диффузионные модели для повышения разрешения видео

Решает проблему временной согласованности через сочетание локального и глобального подходов

Управление процессом генерации:

- **Текстовые подсказки** позволяют влиять на стиль и детали
- **Настройка уровня шума** для баланса между восстановлением и генерацией

Механизмы Upscale-A-Video для временной согласованности



Локальная согласованность:

- Обучение U-Net с временными слоями и интеграцией межкадровой информации
- Файнтюнинг VAE-декодера с учётом временных связей для уменьшения низкоуровневых мерцаний

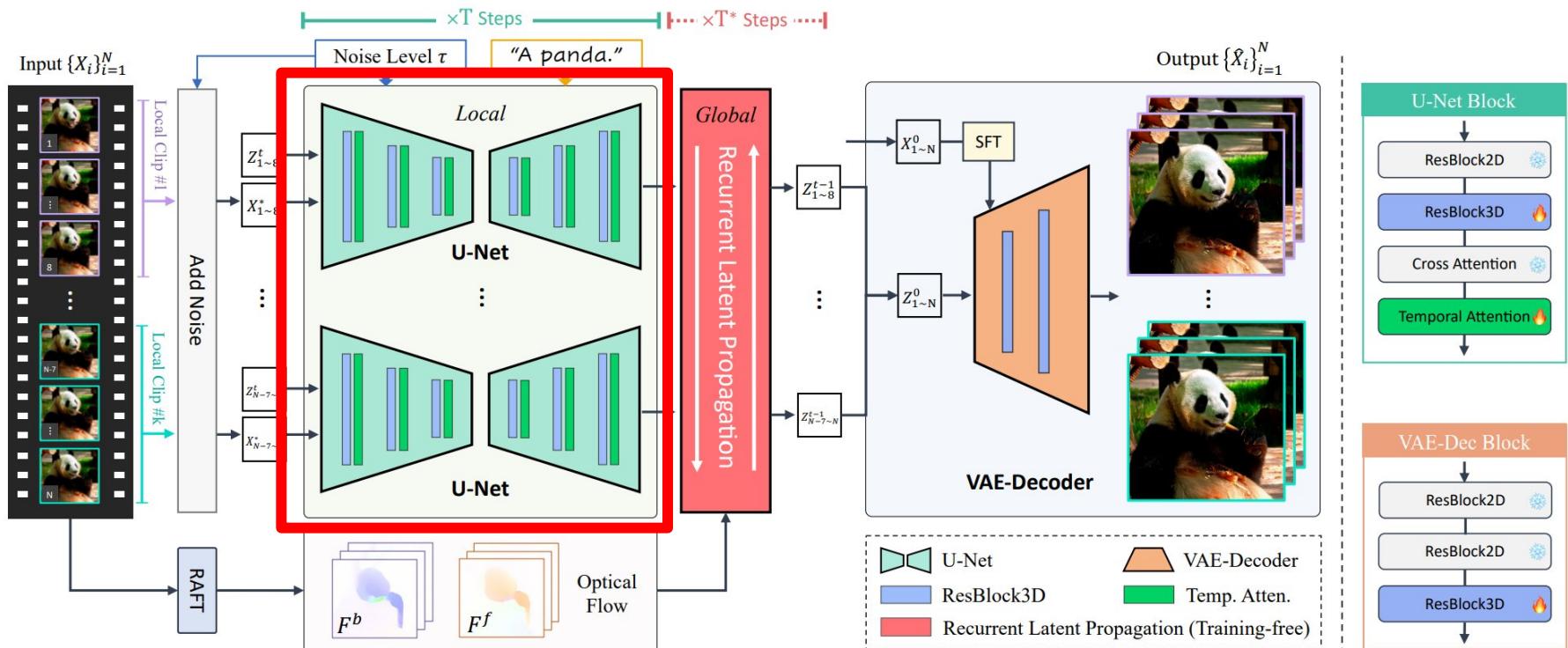
Глобальная согласованность (**Рекурсивная латентная пропагация**):

- Использует оптический поток для передачи информации между кадрами без обучения дополнительных параметров
- Обеспечивает согласованность на уровне всего видео

Механизмы Upscale-A-Video для временной согласованности

Локальная согласованность:

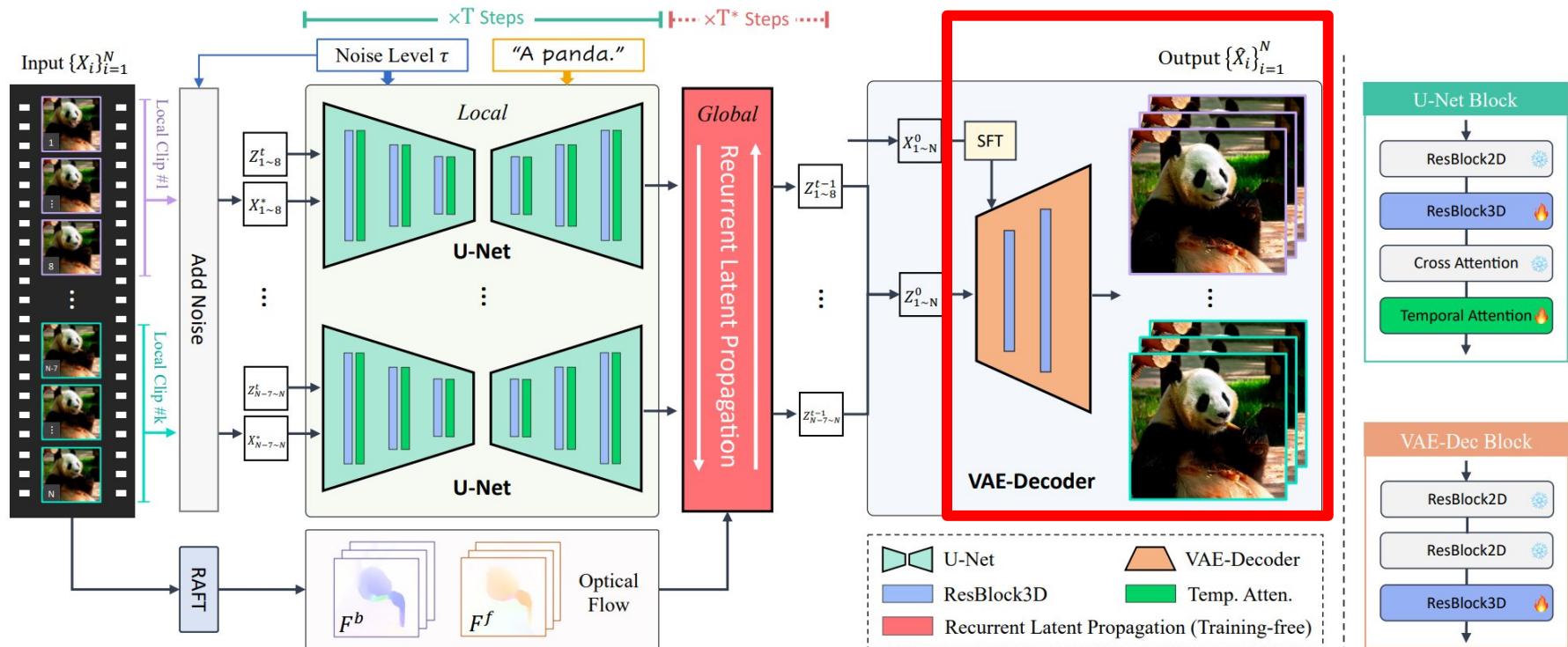
- Обучение U-Net с временными слоями и интеграцией межкадровой информации



Механизмы Upscale-A-Video для временной согласованности

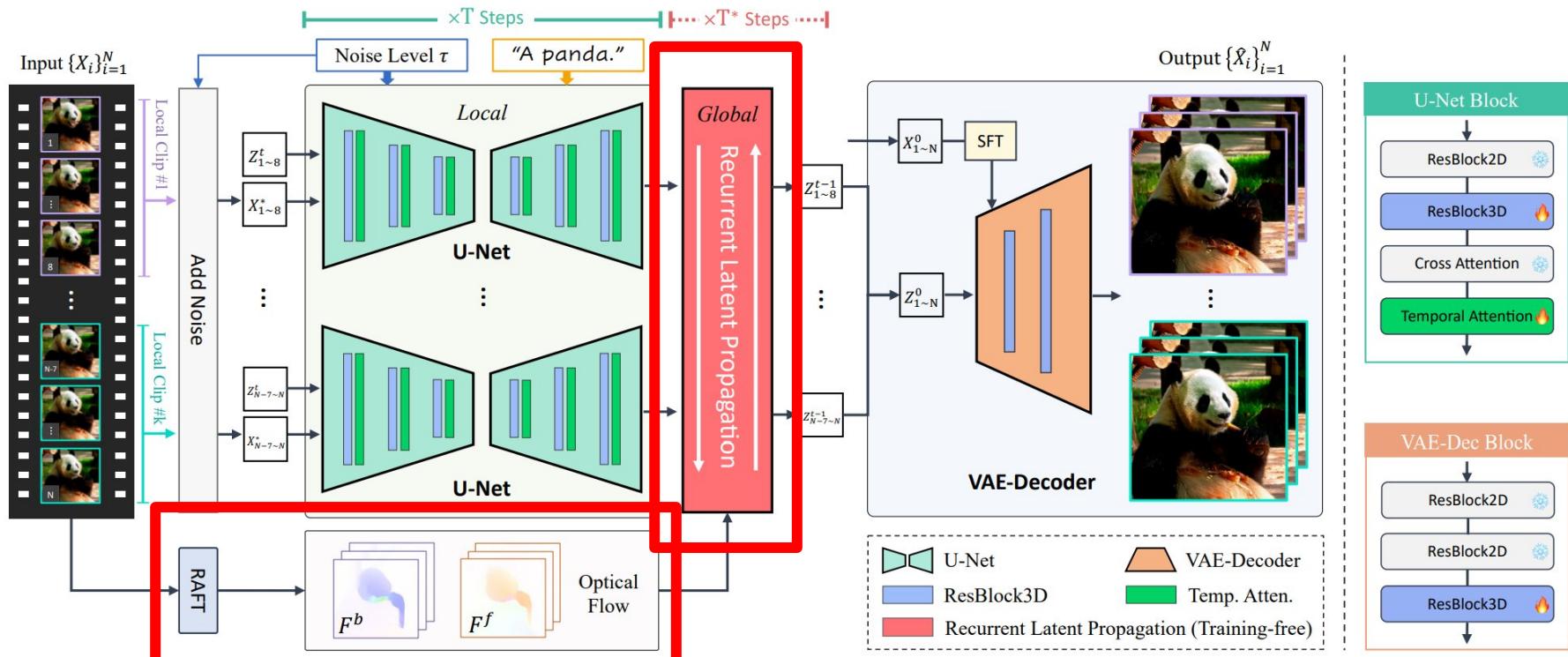
Локальная согласованность:

- Файнтюнинг VAE-декодера с учётом временных связей для уменьшения низкоуровневых мерцаний



Механизмы Upscale-A-Video для временной согласованности

Глобальная согласованность (Рекурсивная латентная пропагация): Использует оптический поток для передачи информации между кадрами без обучения дополнительных параметров



Преимущества и недостатки диффузионных моделей в SR



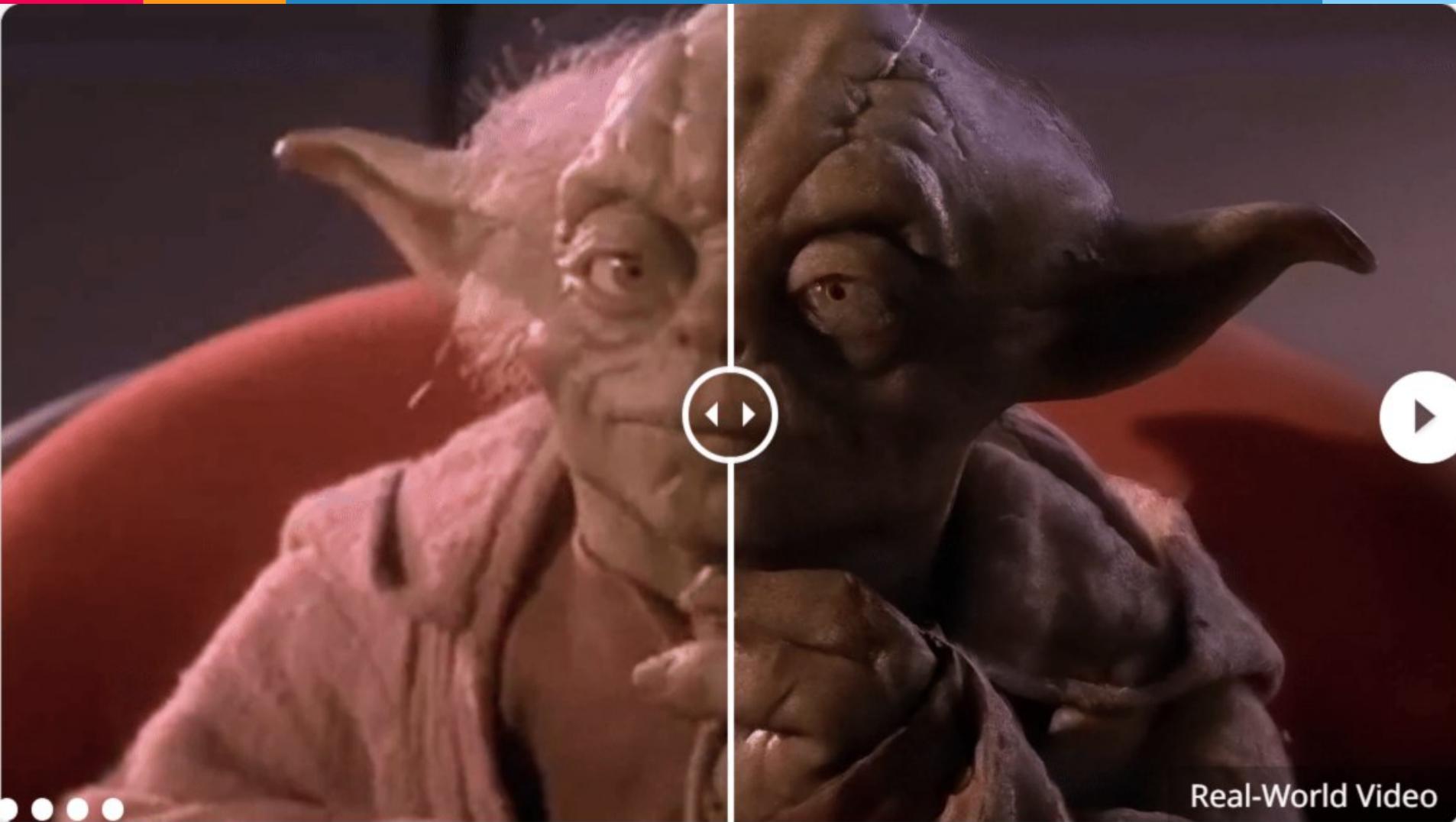
Преимущества:

- Генерация высококачественных и реалистичных деталей в изображениях
- Гибкость в управлении процессом генерации

Недостатки:

- Требуют больших вычислительных ресурсов и времени на генерацию
- Сложность в обучении и настройке

Примеры результатов Upscale-A-Video



Сравнение современных методов SR



Сверточные сети (BasicVSR)

- **Преимущества:** быстрые и эффективные
- **Недостатки:** могут не восстанавливать мельчайшие детали

Генеративные модели (VideoGigaGAN):

- **Преимущества:** способные восстанавливать мельчайшие детали и текстуры
- **Недостатки:** требовательны к вычислительным ресурсам

Трансформеры (VRT):

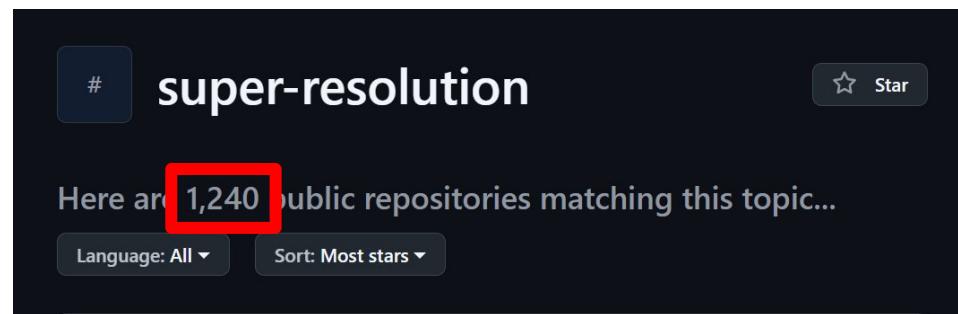
- **Преимущества:** моделируют долгосрочные временные зависимости
- **Недостатки:** высокая вычислительная сложность

Диффузионные модели (Upscale-A-Video)

- **Преимущества:** высокое качество и гибкость в генерации
- **Недостатки:** требуют больше вычислительных ресурсов, сложнее в обучении

Заключение

Нейросетевые методы значительно продвинули SR видео
Каждый подход имеет свои сильные и слабые стороны и
подходит для определённых задач
Область продолжает активно развиваться



Содержание

- Введение
- Так ёщё мой отец и мой дед делали
- Смартфоны — двигатель прогресса
- Примеры нейросетевых архитектур
- **Лаборатория и SR**
 - **ImageSR Benchmark**
 - VideoSR Benchmark
 - SR Metrics Benchmark
 - SR for Video Compression Benchmark

ImageSR Benchmark

Основная идея бенчмарка

- На гитхабе сотни алгоритмов Super Resolution, и их число быстро увеличивается. Но пока никто не сделал масштабное сравнение. Наша задача – понять, какие алгоритмы работают лучше всех
- Упор бенчмарка на красоте картинки. Мы не будем штрафовать алгоритмы за недостоверное восстановление деталей, если получается приятный глазу результат

Стандартные метрики не работают



ESRGAN_PSNR

PSNR: 25.49

SSIM: 0.78



ggpxl-compr

PSNR: 23.45

SSIM: 0.71

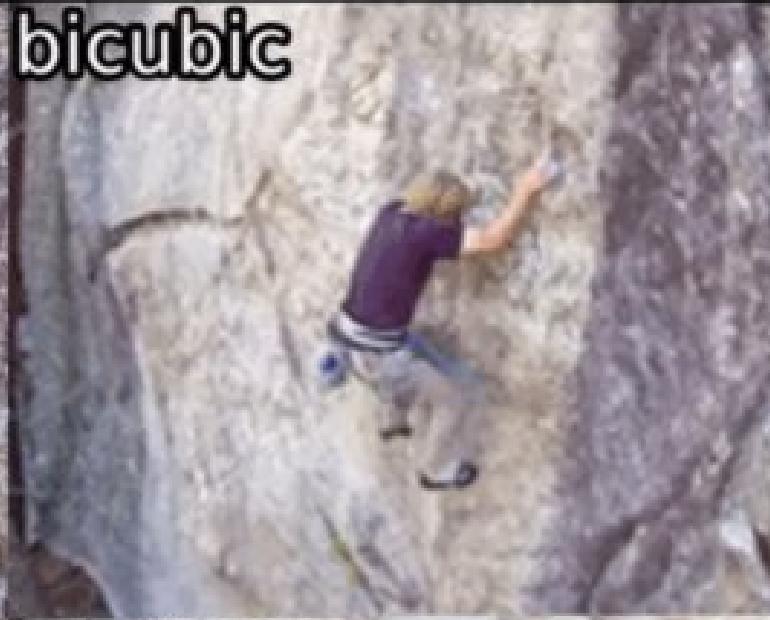
Различные артефакты (1)

Part 1/1 of climbers 4x bicubic; frame 1

GT



bicubic



Topaz ahq-11



RealSR



Различные артефакты (2)

Part 1/1 of dogs 4x bicubic; frame 1

GT



bicubic



Topaz ahq-11



RealSR

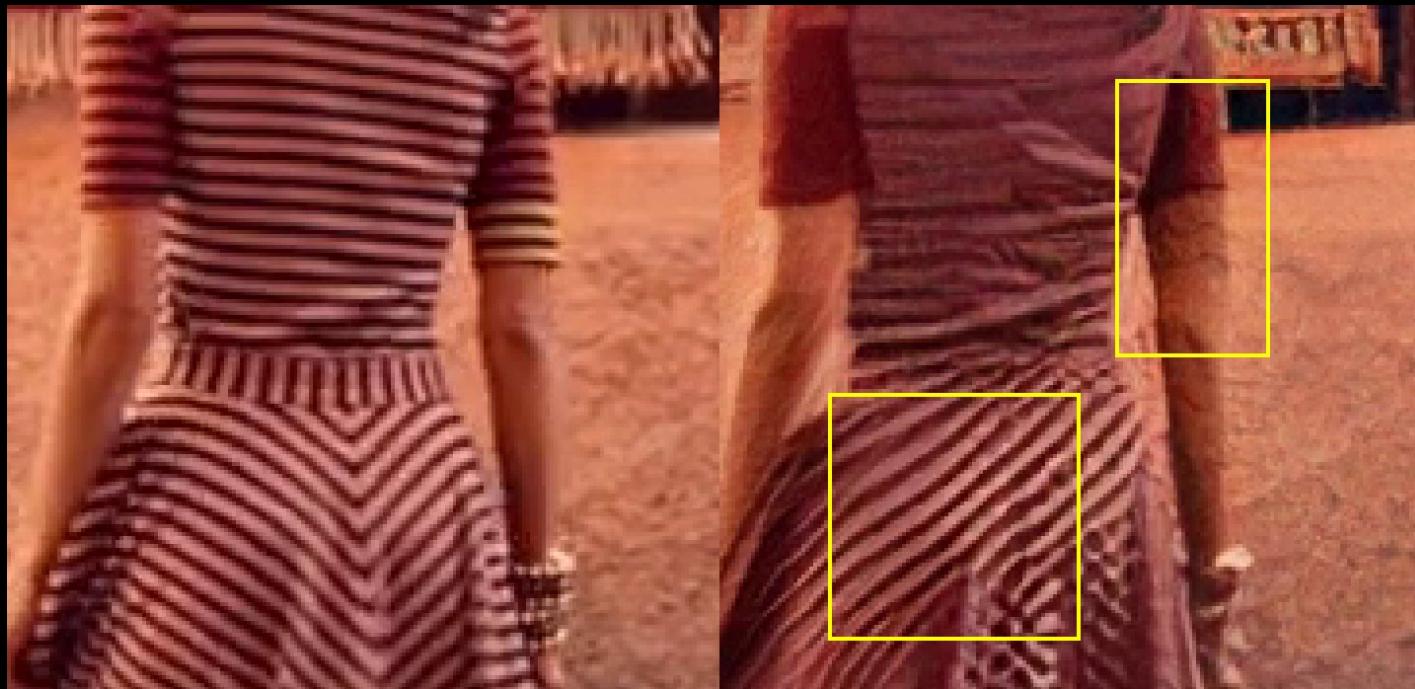


Различные артефакты (3)

Part 1/1 of dress 4x bicubic; Global worst psnr of Topaz ahq-11; frame 1



Различные артефакты (4)



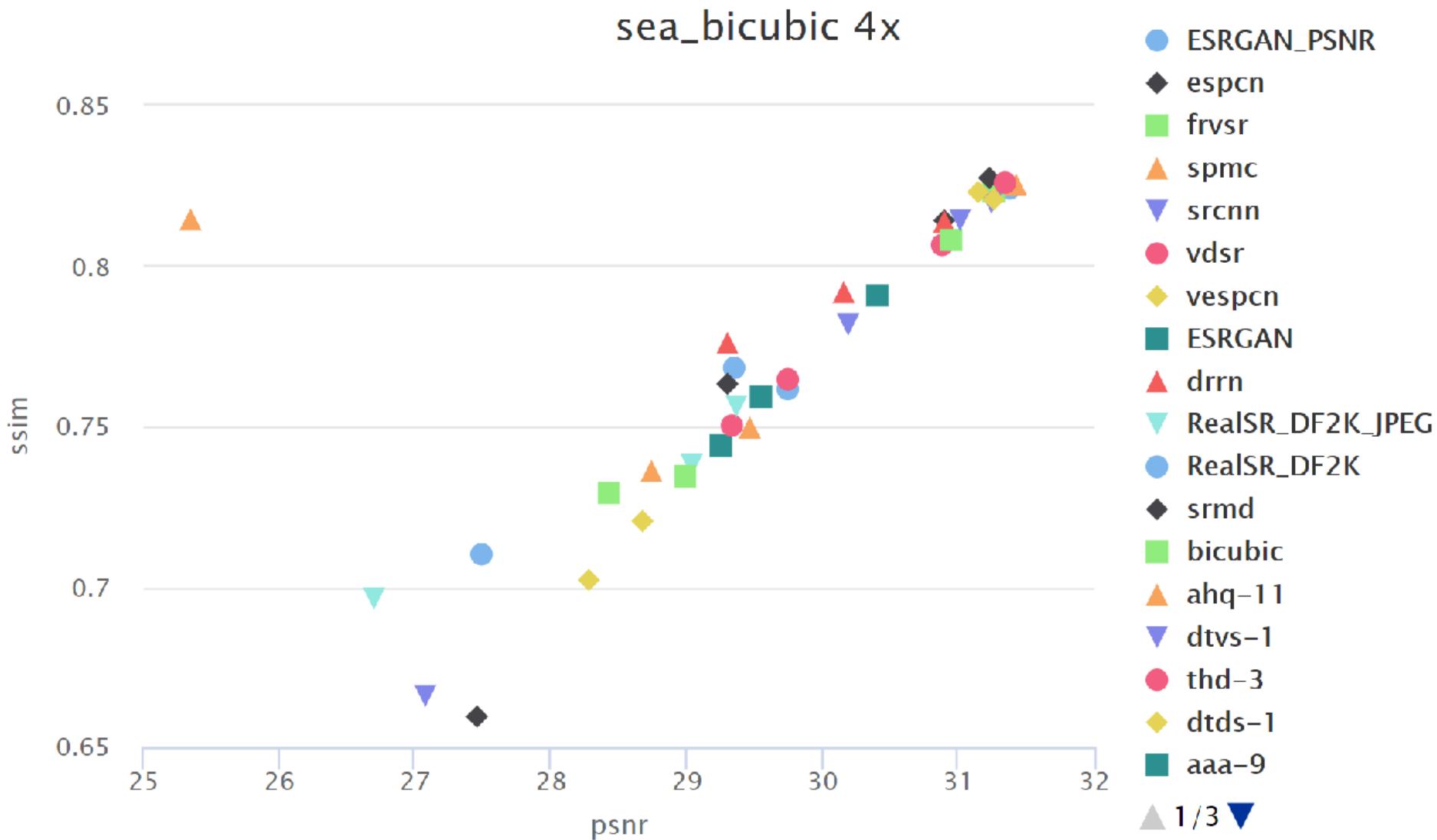
Муар (видео)



Визуализация PSNR



Прогнали 36 моделей на 33 видео



Video Upscalers Benchmark

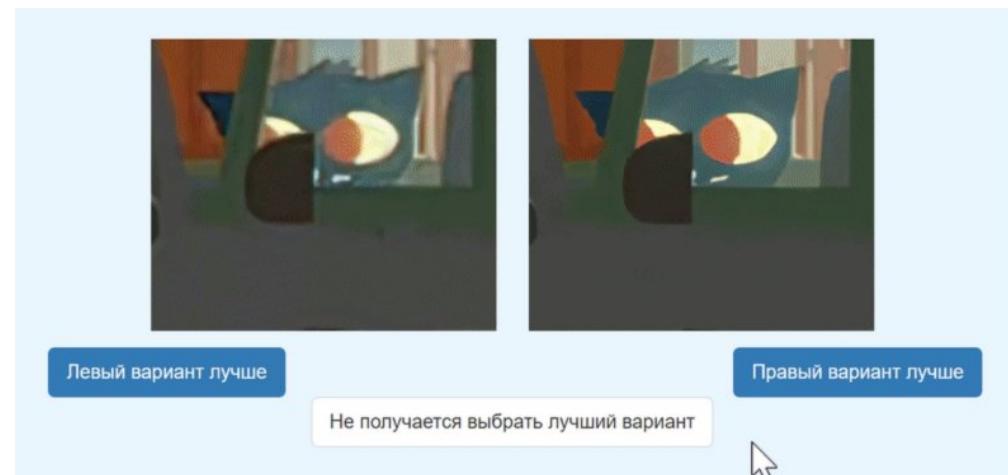
- Крупнейшее сравнение методов SR для улучшения визуального качества видео
- Анализ работы на съемке и мультиплексии
- Тестирование x4 и x2 SR

videoprocessing.ai/benchmarks/video-upscalers.html



Video Upscalers Benchmark

- **3700+** участников субъективных сравнений
- **Side-by-side** сравнение (**Subjectify.us**)
- **30 сегментов** с различными типами контента и легким сжатием, чтобы отразить поведение апскейлеров в реальных условиях



Содержание

- Введение
- Так ёщё мой отец и мой дед делали
- Смартфоны — двигатель прогресса
- Примеры нейросетевых архитектур
- **Лаборатория и SR**
 - ImageSR Benchmark
 - **VideoSR Benchmark**
 - SR Metrics Benchmark
 - SR for Video Compression Benchmark

Video Super Resolution Benchmark

Основная идея бенчмарка



- Video Super Resolution очень популярное пространство для исследований. Сейчас не существует объективной метрики, которая бы адекватно отражала возможность алгоритма восстанавливать реальные детали
- **Основная цель бенчмарка** – создание такой метрики
- Предполагается, что метрика будет отражать **именно качество восстановления деталей**, нежели общую "красоту" картинки

Video Super Resolution Benchmark

Тестовый стенд



Тестовый стенд

QR-коды



Тестовый стенд

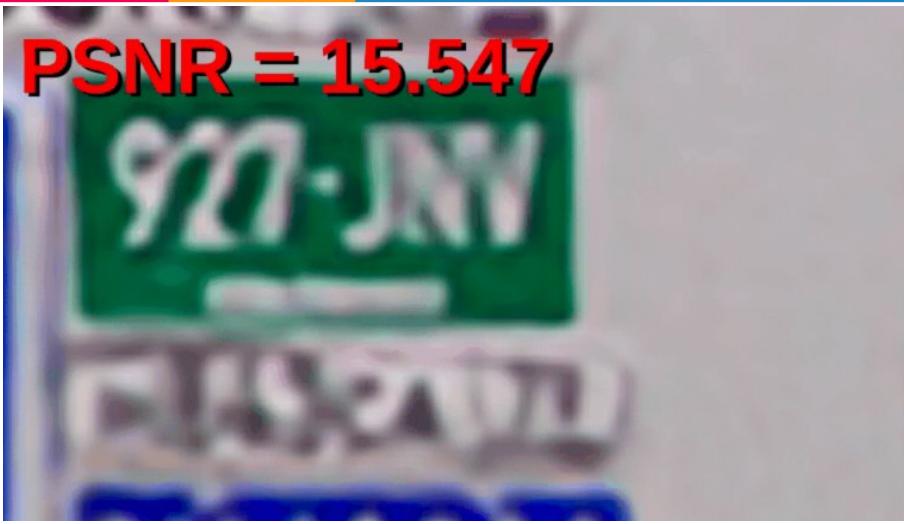
QR-коды



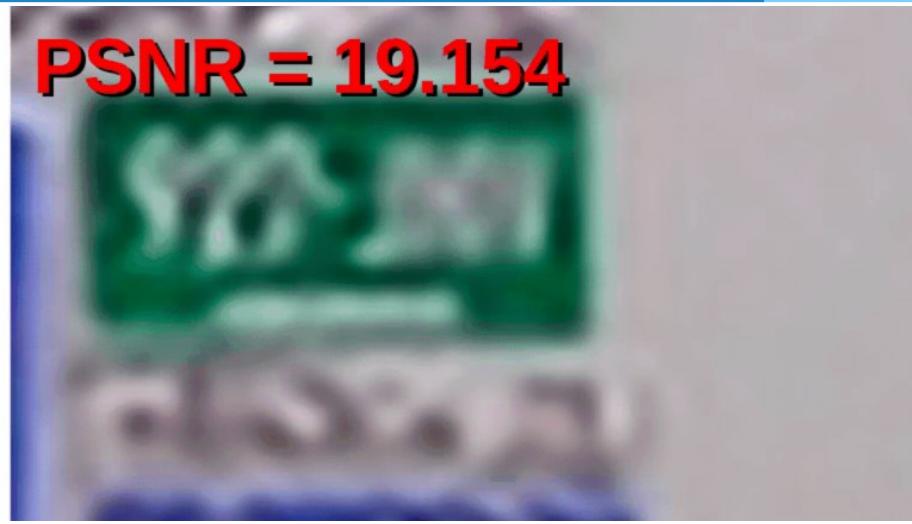
Тестовый стенд

Автомобильные номера

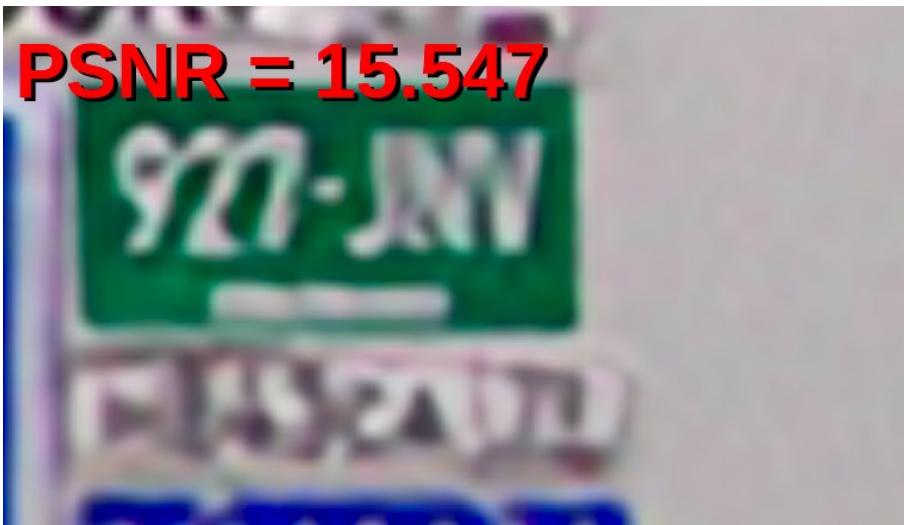
PSNR = 15.547



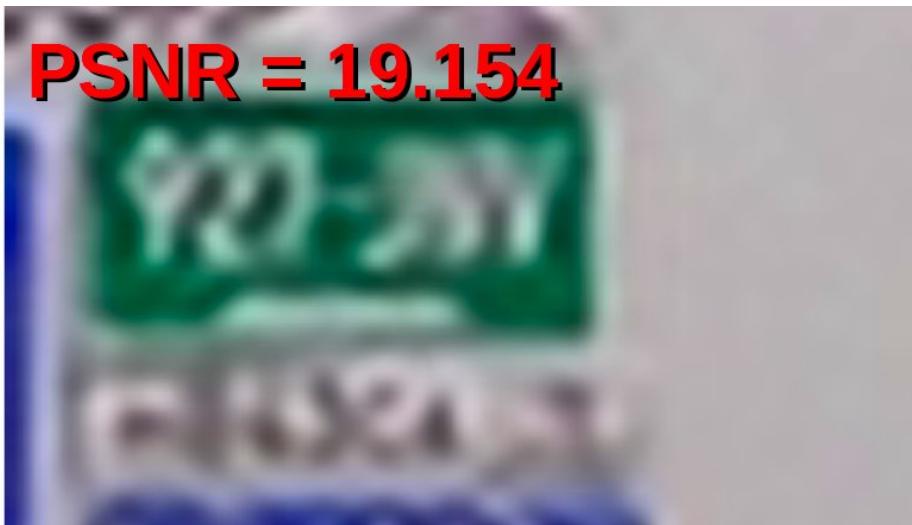
PSNR = 19.154



PSNR = 15.547

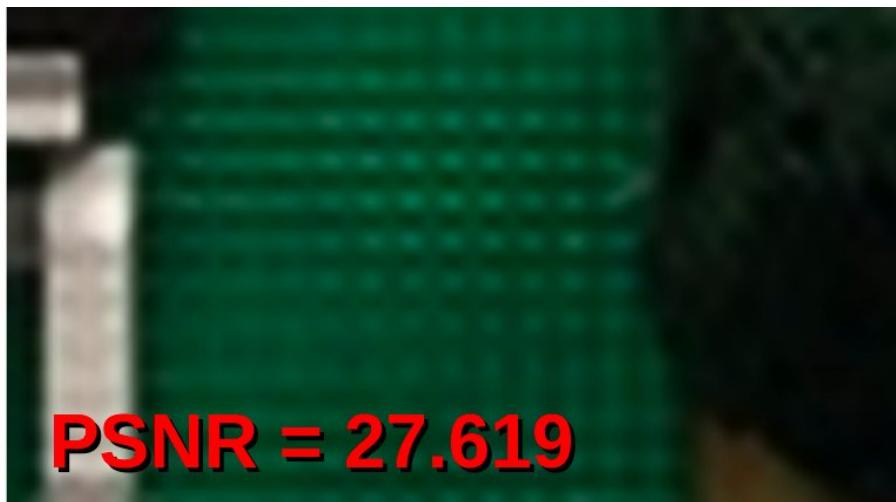
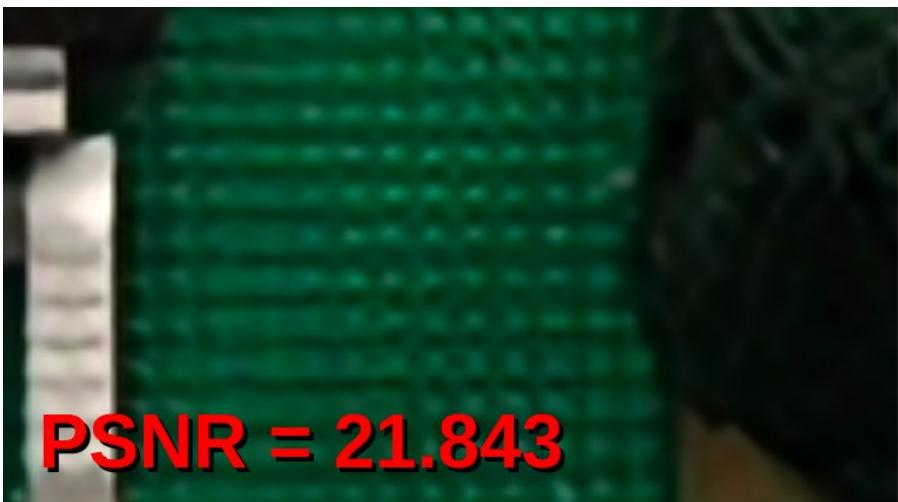
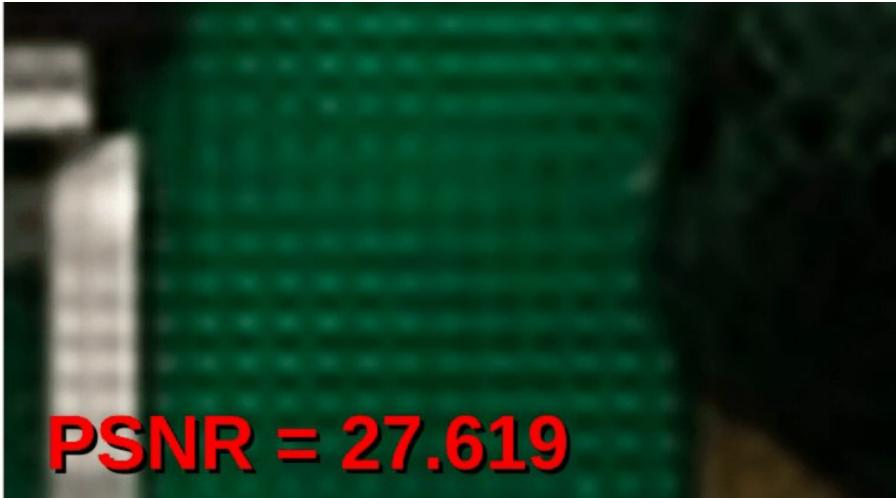
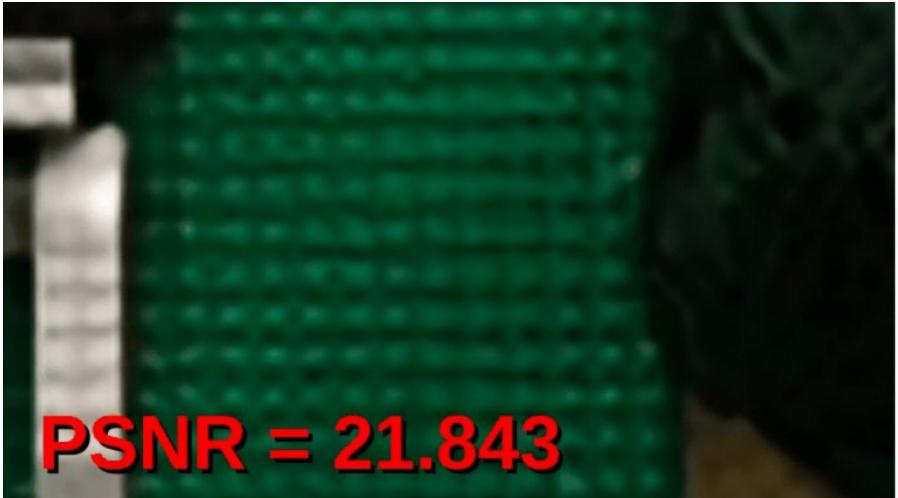


PSNR = 19.154



Тестовый стенд

Лего



Содержание

- Введение
- Так ёщё мой отец и мой дед делали
- Смартфоны — двигатель прогресса
- Примеры нейросетевых архитектур
- **Лаборатория и SR**
 - ImageSR Benchmark
 - VideoSR Benchmark
 - **SR Metrics Benchmark**
 - SR for Video Compression Benchmark

Super-Resolution Metrics Benchmark



Основная идея бенчмарка

- Современные методы Image и Video Super-Resolution часто не улучшают изображение или видео, а искажают их (особенно если оригинал очень низкого качества). Поэтому хотелось бы иметь объективную метрику, способную корректно оценивать работу SR. Однако как показывает практика, даже SOTA-метрики для простой оценки качества изображения и видео (без привязки к SR) плохо справляются с данной задачей
- Основная цель бенчмарка – нахождение или создание такой метрики**

Super-Resolution Metrics Benchmark



Метрики не справляются

PaQ-2-PiQ_(↑): 56.896

HyperIQA_(↑): 0.287

MUSIQ_(↑): 30.596

PaQ-2-PiQ_(↑): 58.564

HyperIQA_(↑): 0.328

MUSIQ_(↑): 35.054



Super-Resolution Metrics Benchmark

Метрики не справляются



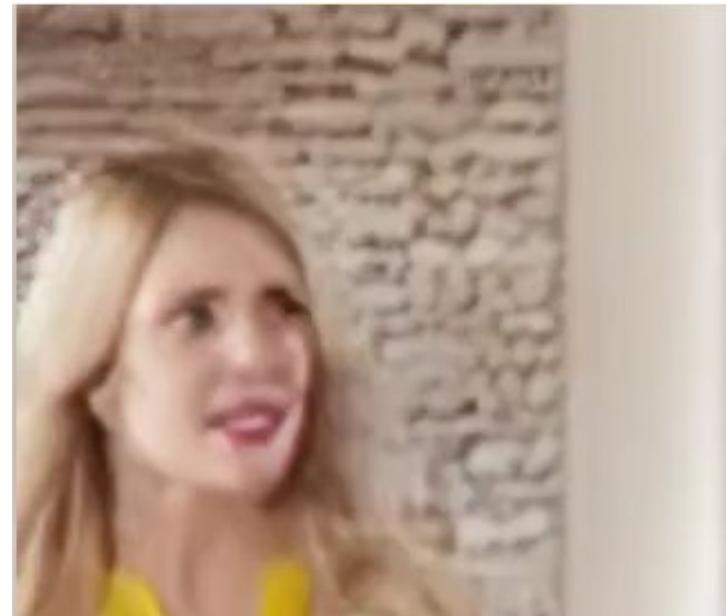
PaQ-2-PiQ: 67.85



Видео с применением

Real-ESRGAN

PaQ-2-PiQ: 56.51



Видео с применением

ggpkl-vcomp

Super-Resolution Metrics Benchmark

Метрики не справляются

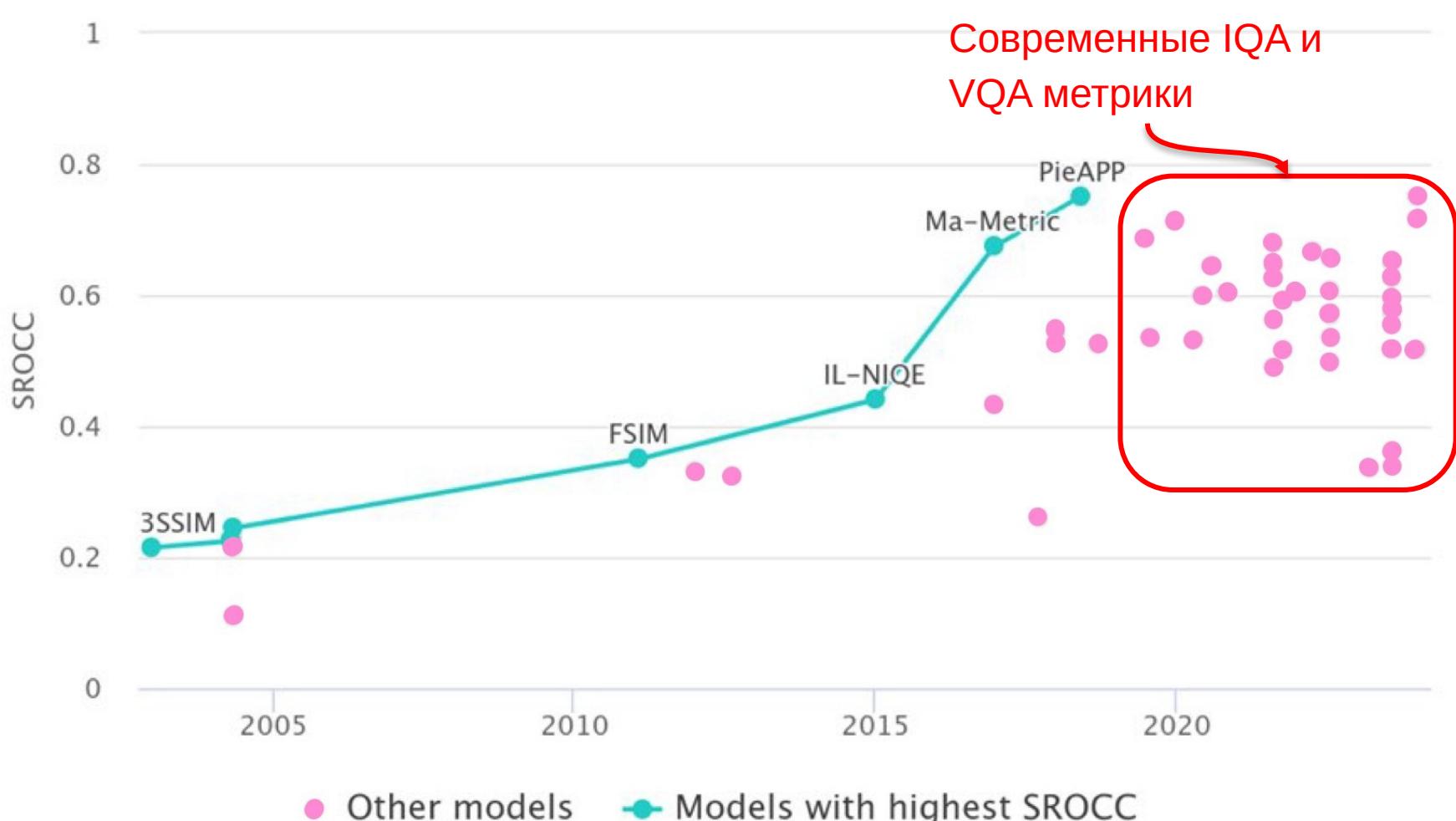


Визуализация наиболее "понравившихся"
метрике PaQ-2-PiQ мест

Super-Resolution Metrics Benchmark

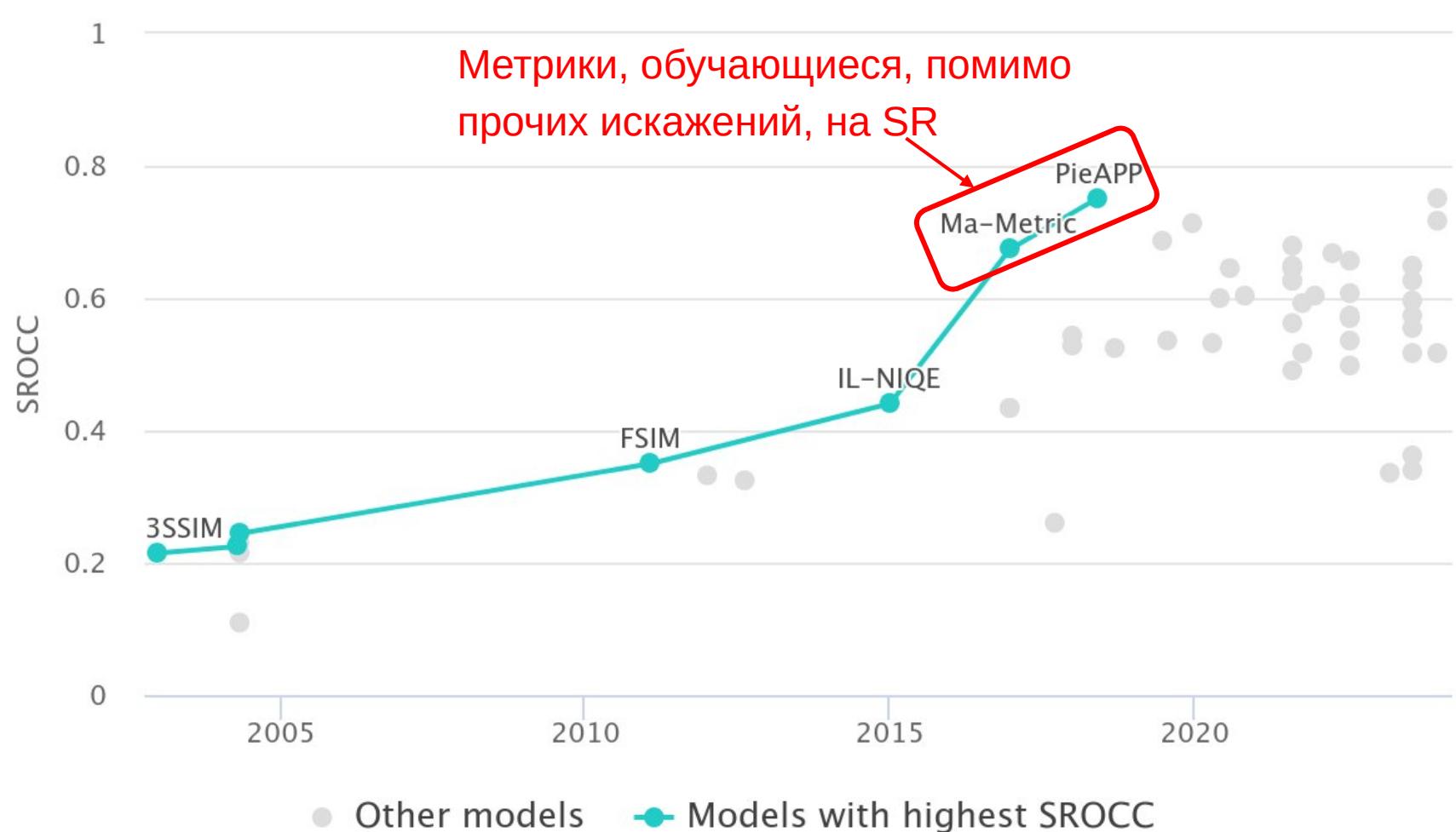


Итоги исследования на настоящий момент



Super-Resolution Metrics Benchmark

Итоги исследования на настоящий момент



Содержание

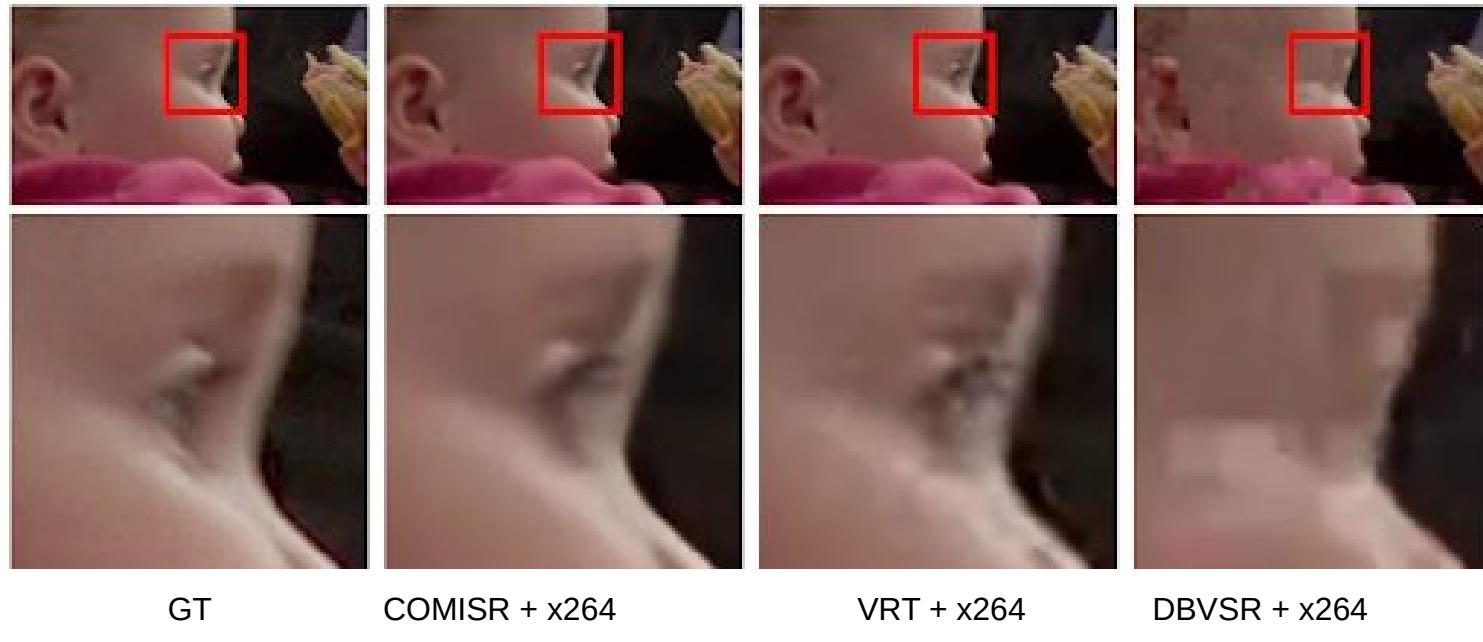
- Введение
- Так ёщё мой отец и мой дед делали
- Смартфоны — двигатель прогресса
- Примеры нейросетевых архитектур
- **Лаборатория и SR**
 - ImageSR Benchmark
 - VideoSR Benchmark
 - SR Metrics Benchmark
 - **SR for Video Compression Benchmark**

SR for Compression Benchmark



Основная идея бенчмарка

Некоторые видеокодеки используют SR во время сжатия, однако в таком случае могут появляться дополнительные артефакты.



GT

COMISR + x264

VRT + x264

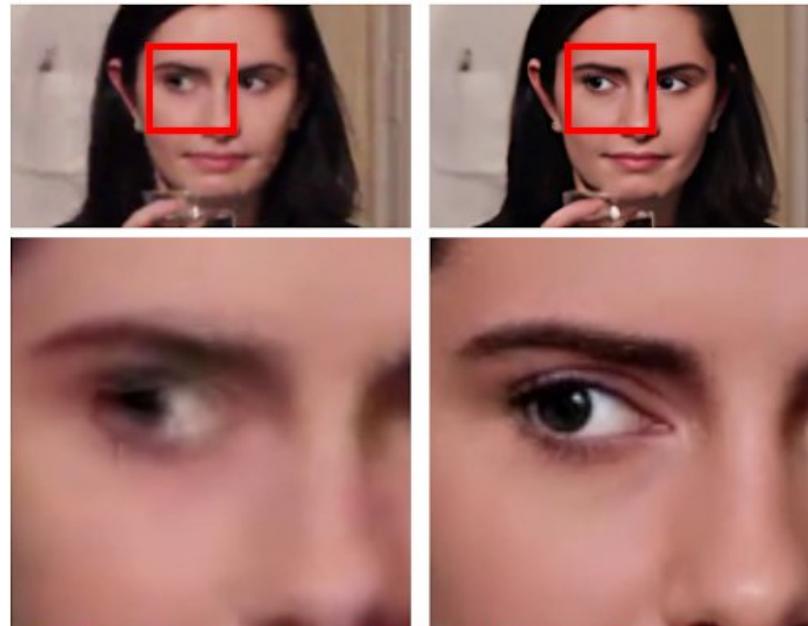
DBVSR + x264

SR for Compression Benchmark

Основная идея бенчмарка

Многие метрики качества не могут корректно определять такие артефакты.

PSNR: 26.908
SSIM: 0.955
VMAF: 81.723
LPIPS: 0.311



PSNR: 24.911
SSIM: 0.937
VMAF: 95.388
LPIPS: 0.332

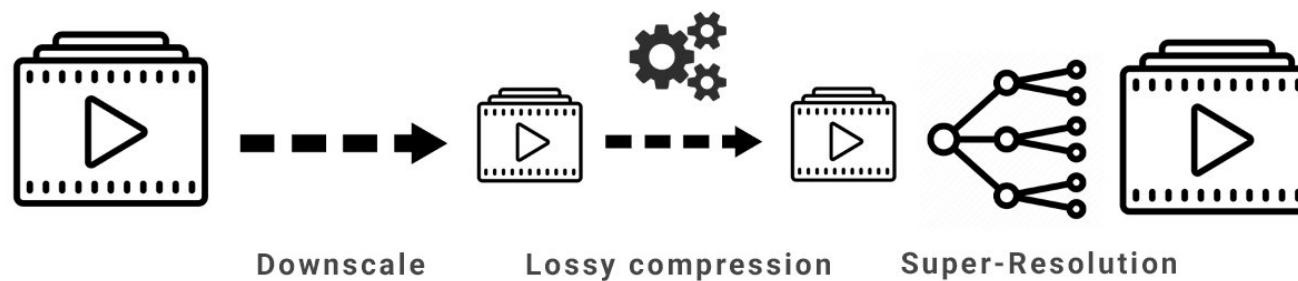
SR for Compression Benchmark

Устройство бенчмарка



Бенчмарк состоит из:

- **5** стандартов кодеков
(H.264, H.265, H.266, AV1, AVS3)
- **90** пар SR+codec
- 9 тестовых видео
- **6** уровней сжатия
- Субъективных оценок от **5300+** участников



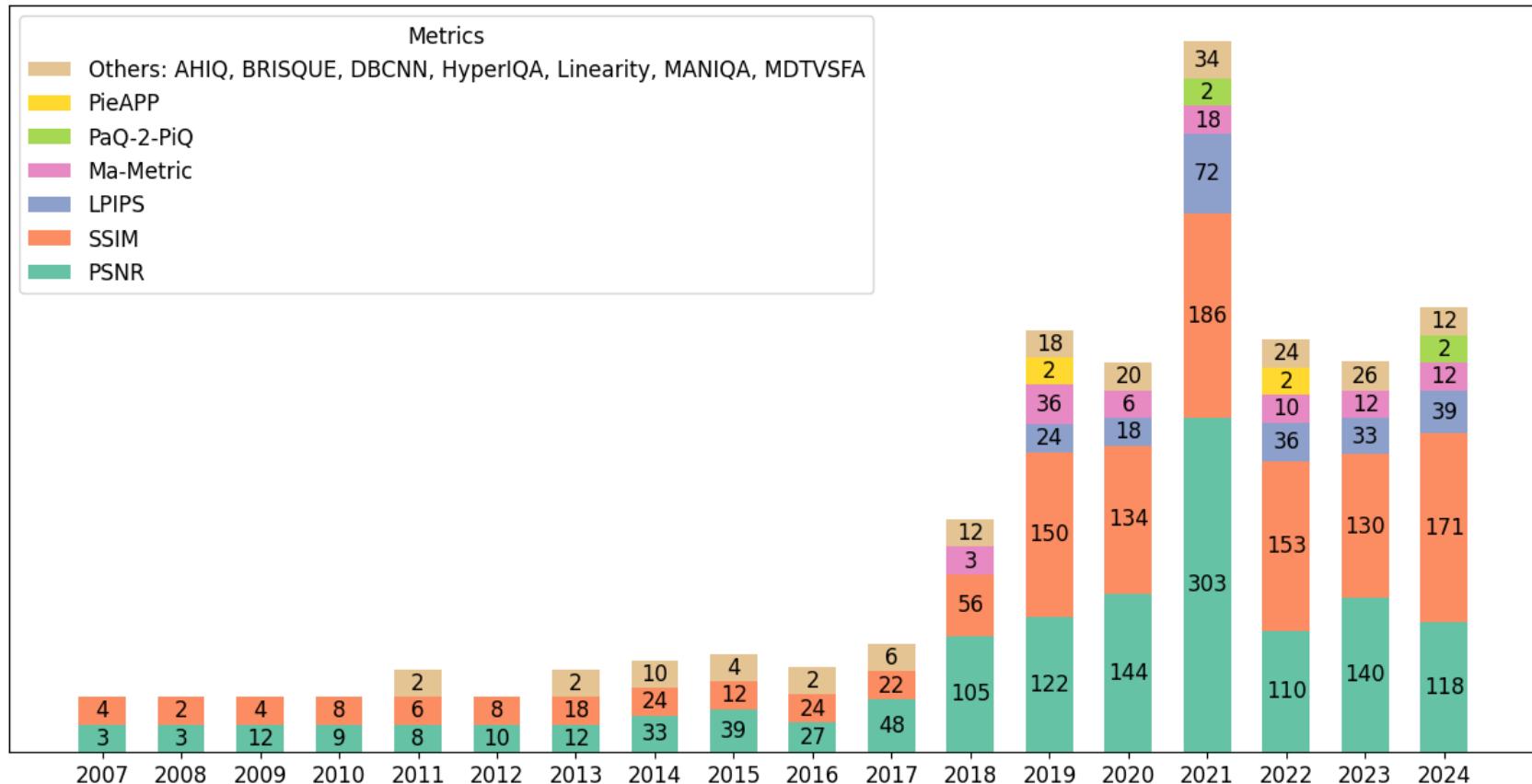
SR for Compression Benchmark

PSNR до сих пор популярен



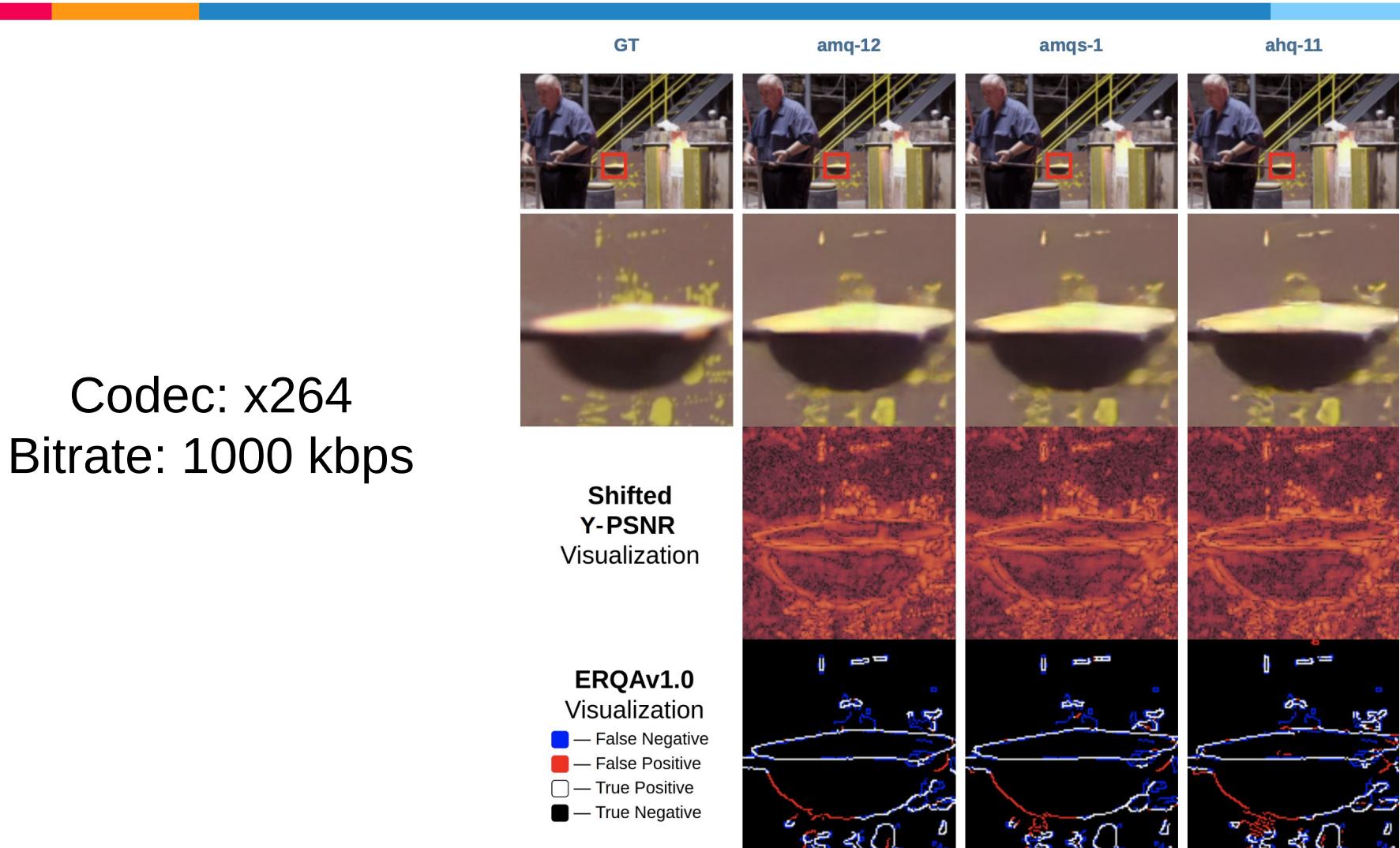
Mentions of Metrics in Papers with "Super-Resolution" keyword

Based on 1494 Papers



SR for Compression Benchmark

Результаты (лучшие методы)

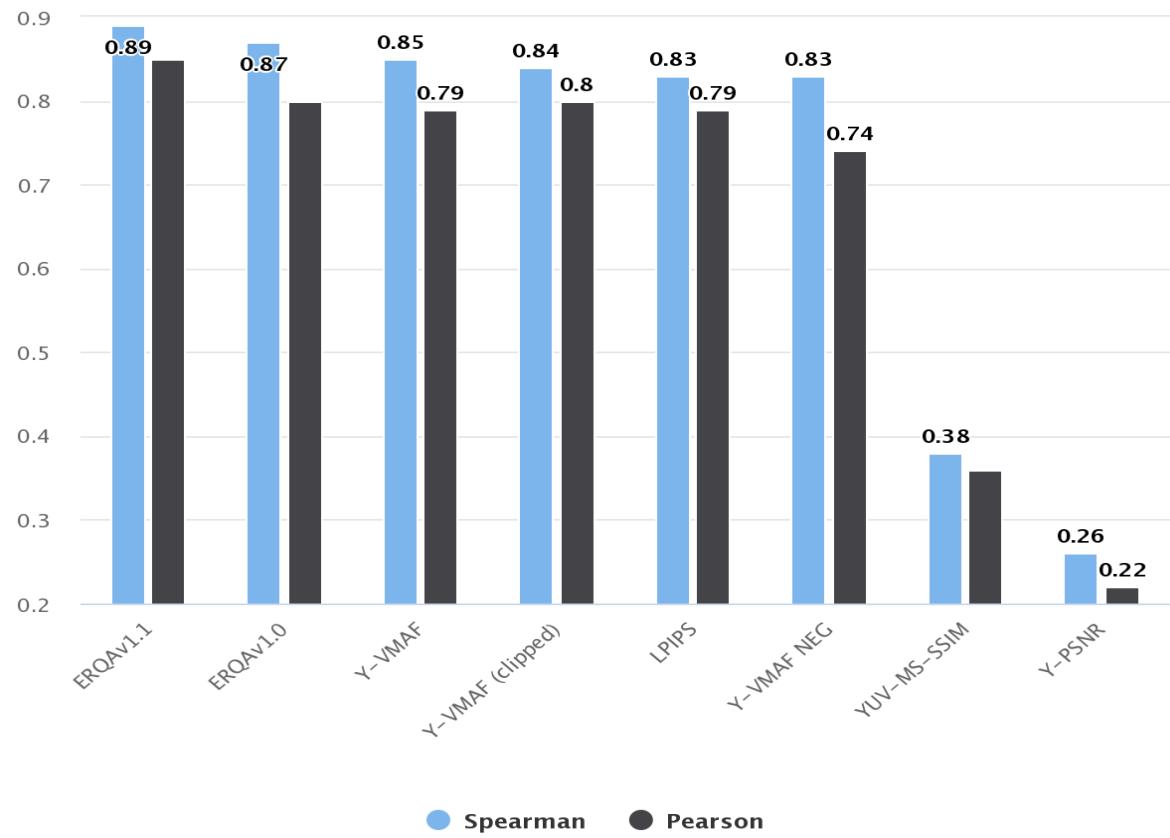


SR for Compression Benchmark

Корреляции объективных и субъективных оценок

Для субъективного сравнения:

- x264 кодек
- 3 различных битрейта
- 2 кропа с разрешением 320×270 для каждого видео

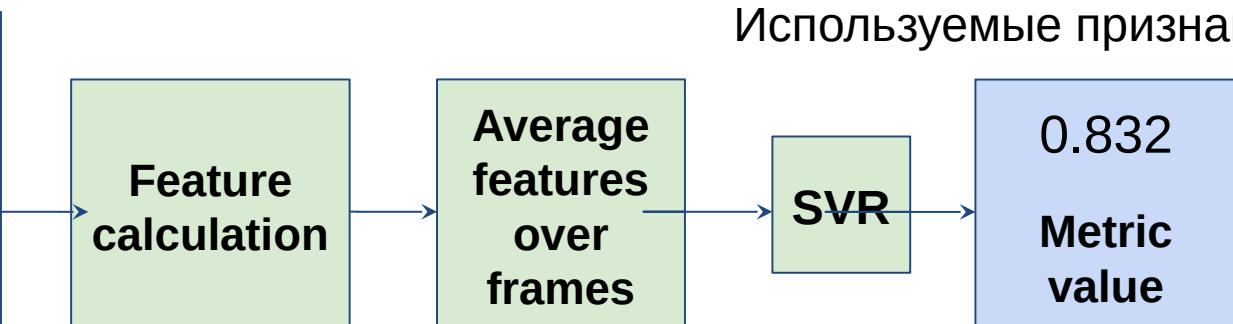
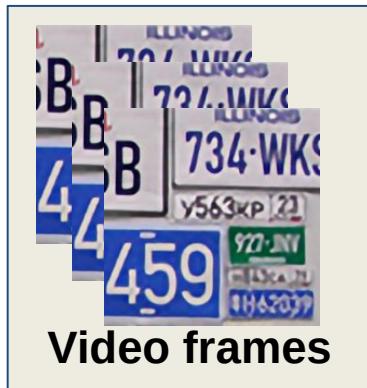


SR for Compression Benchmark

Предложенная метрика

- Мы использовали SVR для предсказания значения метрики.
- Также мы использовали некоторые другие метрики качества видео в качестве признаков модели.

Feature	Description
MDTVSFA	<i>MDTVSFA_value</i>
ERQA	<i>ERQA_value</i>
LPIPS	<i>LPIPS_value</i>
ERQAxMDTVSFA	$MDTVSFA_value \times ERQA_value$
ERQAxLPIPS	$(1 - LPIPS_value) \times ERQA_value$
SI/TI	Google SI/TI [29] features
Colorfulness	Mean colorfulness [13] of all frames
Bitrate	The bitrate of a video



Архитектура предложенной метрики

SR for Compression Benchmark

Предложенная метрика



Наша метрика показывает лучшие корреляции с субъективными оценками на предложенном датасете

Metric	PLCC	SRCC
MS-SSIM	0.146	0.151
PSNR	0.187	0.285
VMAF	0.344	0.448
LPIPS	0.414	0.431
ERQA	0.582	0.624
MDTVSFA	0.634	0.644
ERQAxMDTVSFA	0.770	0.801
Proposed	0.821	0.838

Корреляции метрик на нашем датасете

Prediction Model	Wireless	H.264	MPEG-2	All Data
PSNR	0.4334	0.4296	0.3588	0.3684
SSIM	0.5233	0.6514	0.5545	0.5257
MS-SSIM	0.7285	0.7051	0.6617	0.7361
Speed SSIM	0.5630	0.7086	0.6185	0.5849
VSNR	0.7019	0.6460	0.5915	0.6755
VQM	0.7214	0.6520	0.7810	0.7026
V-VIF	0.5507	0.6807	0.6116	0.5710
Spatial MOVIE	0.7928	0.7066	0.6911	0.7270
Temporal MOVIE	0.8114	<u>0.7797</u>	0.8170	0.8055
MOVIE	<u>0.8109</u>	<u>0.7664</u>	<u>0.7733</u>	<u>0.7890</u>
Proposed	<u>0.7934</u>	0.8465	<u>0.7554</u>	<u>0.7946</u>

Корреляции метрик на датасете Live Video Quality Database

Влияние на область SR

- Независимые, открытые бенчмарки
- Сравнение алгоритмов на множестве тест-кейсов
 - Уникальные датасеты
 - Видео с разнообразным контекстом и семантикой
 - Самые сложные случаи
- Всесторонний анализ слабых мест существующих алгоритмов

Задание SR

GT



Nearest Neighbor (LQ)



Bicubic



Задание выполняется в Jupyter Notebook

- Необходимо реализовать несколько алгоритмов 1D-интерполяции
- Запустить несколько популярных нейросетей для SR
- Разработать классификатор областей с артефактами

Баллы за задание = Линейно отмасштабированная до 15-ти сумма баллов в ноутбуке