

# Эффективная модификация алгоритма адаптивной медианной фильтрации цифровых изображений

Яиков Рафаэль Равильевич

Ярославский государственный университет им. П. Г. Демидова

# Какие бывают шумы?

- \* Аддитивный шум
- \* Мультипликативный шум
- \* Импульсный шум



Оригинал



Гауссов шум



Соль и перец



Импульсный шум

# Подавление шумов

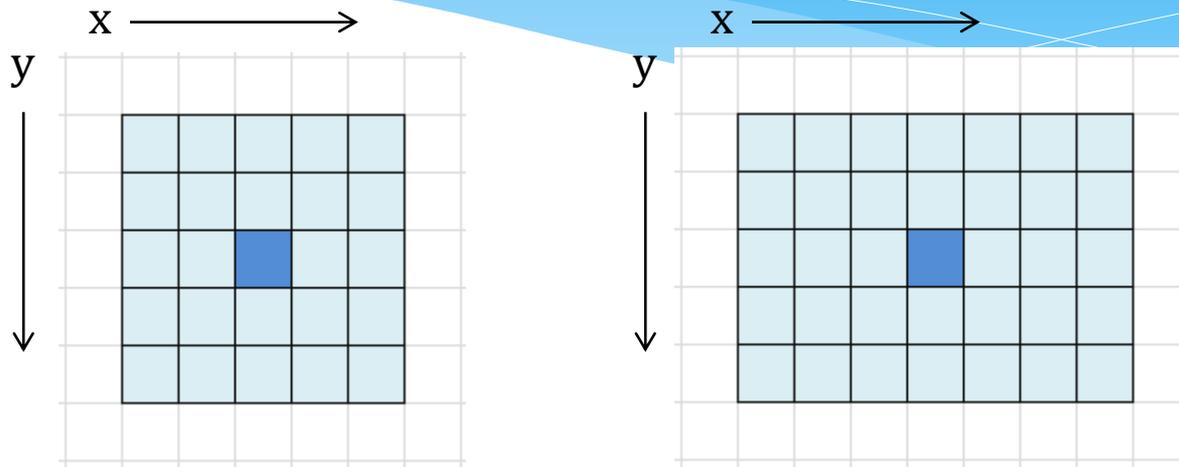
## **Линейные фильтры:**

- \* Скользящие среднее в окне
- \* Фильтр Гаусса и др.

## **Нелинейные фильтры:**

- \* Фильтры на основе ранговой статистики: экстремальные, медианные и др.
- \* Фильтр Винера
- \* Гомоморфная фильтрация и др.

# Принцип линейных фильтров



$$C_{new}(x, y) = \sum_{k=0}^m \sum_{l=0}^n A_{k,l} C_{old}(x - m/2 + k, y - n/2 + l)$$

$C_{new}$ ,  $C_{old}$  – новые и старые значения пикселей изображения соответственно;  
 $A_{k,l}$  – коэффициент, определяющий эффект, который накладывает фильтр;  
 $m$ ,  $n$  – константы, задающие размер окна

# Фильтрация методом свёртки

Вычисление среднего арифметического:

$$f(x,y) = \frac{1}{m \times n} \sum_{(s,t) \in S_{xy}} g(s,t)$$

$S_{xy}$  – прямоугольная окрестность с центром  $x, y$

$g(s, t)$  – значение пикселя оригинала

$f(x,y)$  – значение обработанного пикселя

При выполнении в линейной форме коэффициенты  $A_{k,l} = \frac{1}{m \times n}$

Маска усредняющего фильтра  $3 \times 3$ :

$$\frac{1}{9} \times \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

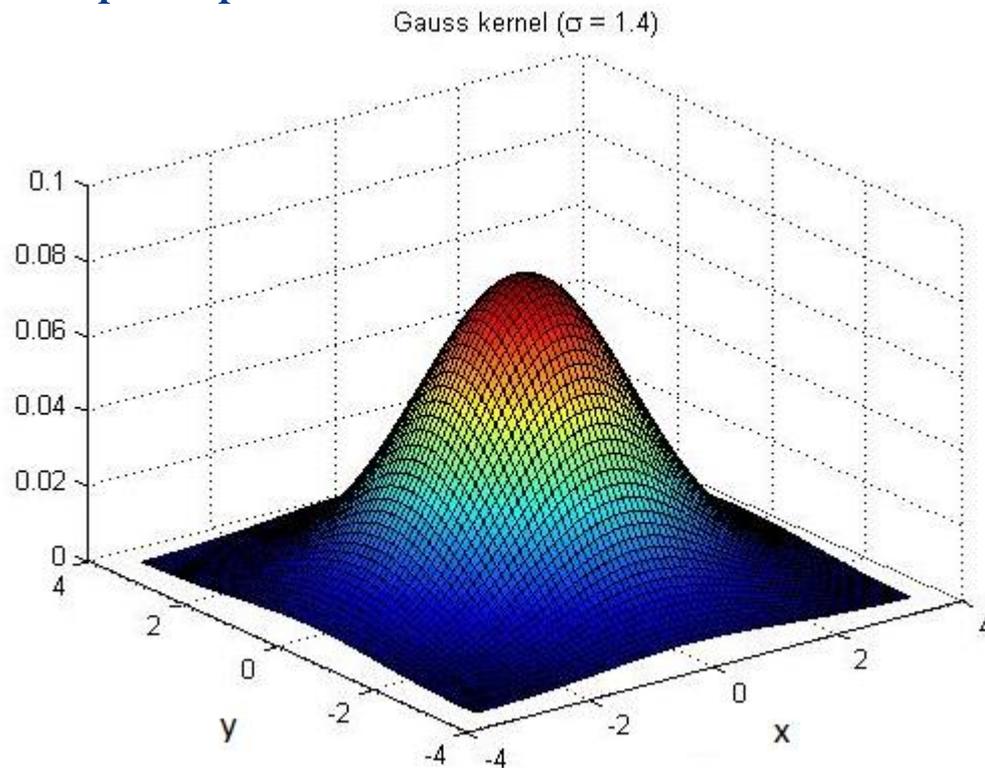
# Фильтрация методом свёртки

Функция распределения коэффициентов фильтра Гаусса в двумерном пространстве принимает вид:

$$A(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

Гауссовская маска 3×3:

$$\frac{1}{16} \times \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$



# Фильтры на основе ранговой статистики

Экстремальные фильтры:

$$f(x, y) = \max_{(s,t) \in S_{xy}} \{g(s, t)\}$$

$$f(x, y) = \min_{(s,t) \in S_{xy}} \{g(s, t)\}$$

Медианные фильтры:

$$f(x, y) = \text{med}_{(s,t) \in S_{xy}} \{g(s, t)\}$$

$S_{xy}$  – прямоугольная окрестность с центром в точке  $(x, y)$

$g(\dots)$  – значение пикселей оригинального изображения

$f(x, y)$  – значение обработанного пикселя изображения

# Медианная фильтрация – как это работает?

150	150	10	150	150	150	150	150	150	150
150	150	10	150	150	150	150	150	150	150
150	150	10	150	150	150	150	150	150	150
10	10	10	10	10	10	10	10	10	10
150	150	150	150	150	150	150	150	150	150
150	150	150	150	150	150	150	150	150	150
150	150	150	150	150	150	150	150	150	150
150	150	150	150	150	150	150	150	150	150

$$L_c = 0,3 \times R + 0,59 \times G + 0,11 \times B$$

Изображение с двумя линиями и фоном

0	150	10	150	150	150	150	150	150	150
150	150	10	255	150	150	150	150	255	150
150	150	10	150	150	150	150	150	150	150
10	10	10	10	10	10	10	10	10	10
150	150	150	150	150	150	150	150	150	150
150	150	150	150	150	150	150	150	0	150
255	150	150	150	150	255	150	150	150	150
150	150	150	150	150	150	150	150	150	150

Изображение с помехой «соль и перец»

0	150	10
150	150	10
150	150	10

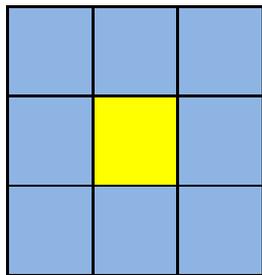


0	10	10	10	150	150	150	150	150
---	----	----	----	-----	-----	-----	-----	-----

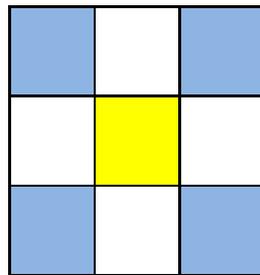


150

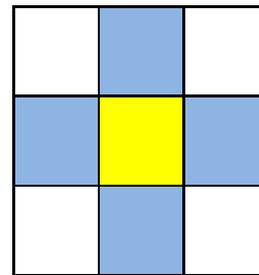
# Варианты ядра фильтров



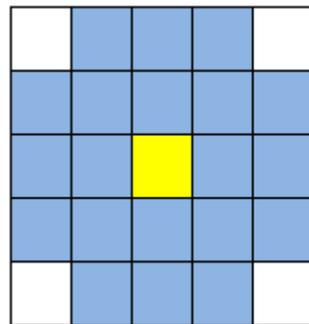
1



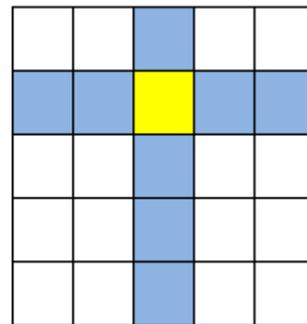
2



3



4



5

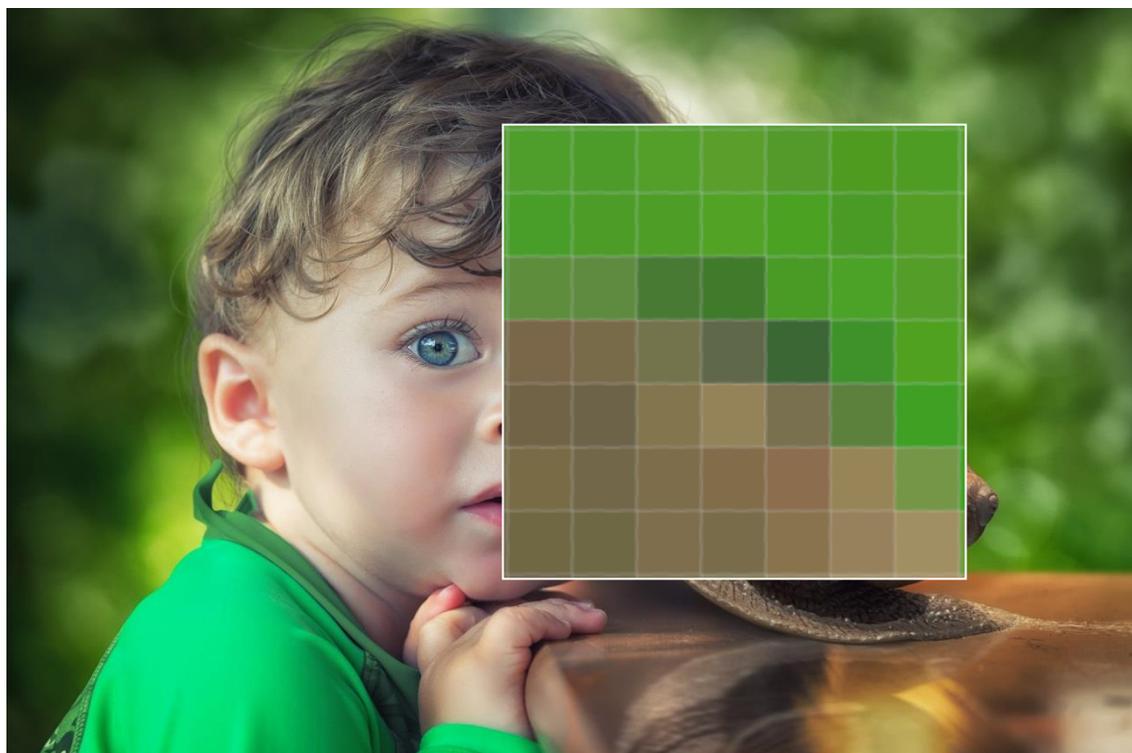
# Адаптивный медианный фильтр

Алгоритм адаптивной медианной фильтрации (АМФ) состоит из двух ветвей.



# Адаптивный медианный фильтр

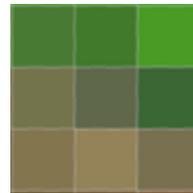
Алгоритм адаптивной медианной фильтрации (АМФ) состоит из двух ветвей.



# Адаптивный медианный фильтр

Первая ветвь:

1. Поиск медианного значения
2. Проверка на адекватность медианы
3. Если медиана исправна, переходим ко второй ветви



Медианна адекватна, если входит во множество:

$$L_{\min} < L_{\text{med}} < L_{\max}$$

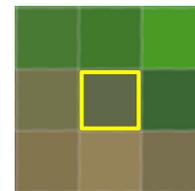
# Адаптивный медианный фильтр

Вторая ветвь:

1. Проверка на адекватность центрального пикселя ядра
2. В случае его испорченности, заменяем на медианное значение, в обратном случае оставляем без изменений

Центральный пиксель адекватен,  
если входит во множество:

$$L_{\min} < L_c < L_{\max}$$

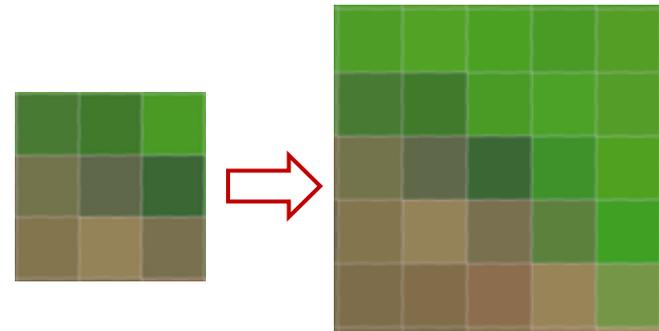


# Адаптивный медианный фильтр

Первая ветвь:

1. Поиск медианного значения
2. Проверка на адекватность медианы
3. Если медиана неисправна, увеличиваем размер фильтрующего окна

В случае, если размер окна меньше либо равен  $S_{max}$ , повторяется первая ветвь, иначе в качестве результата принимается значение  $L_c$



# Модификация алгоритма АМФ

1. Считывание данных из окна обработки (маски фильтра).
2. Применение адаптивной медианной фильтрации.
3. Анализ качества изображения

$$Q = \frac{N}{S}$$

$Q$  – коэффициент, характеризующий качество изображения

$N$  – число исправленных пикселей изображения

$S$  – общее число пикселей

Если  $Q > \sigma$  и число итераций  $n < \text{max}$ , повторить пункт 2

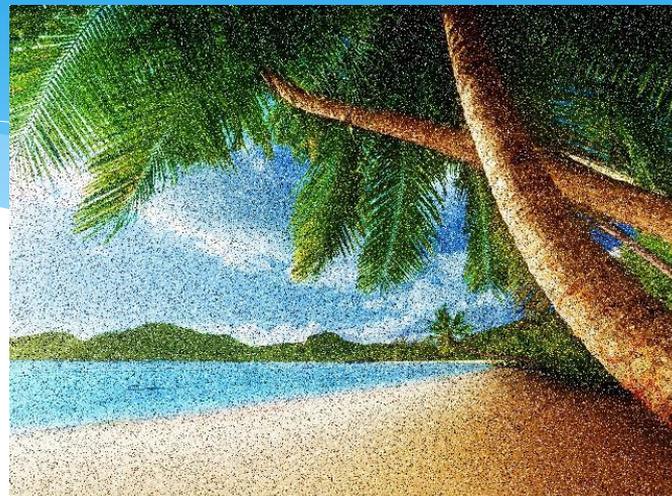
$\sigma$  – пороговое значение шума ( $\approx 0,1313$ )

$\text{max}$  – максимально допустимое число проходов (экспериментально получилось  $\approx 8$ )

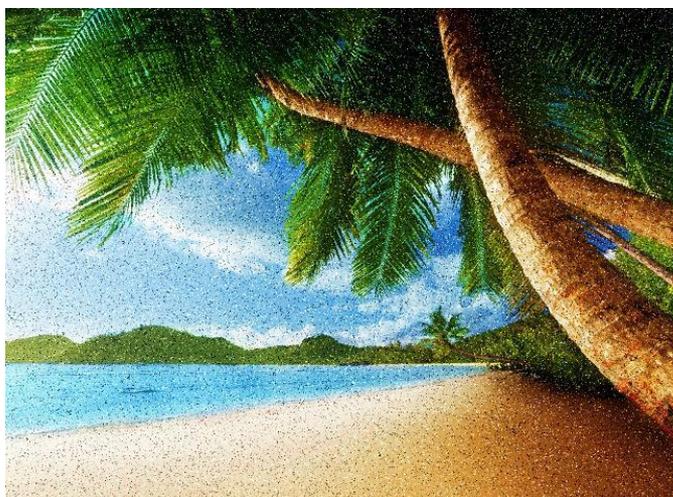
# В чём же отличие?



Исходное изображение



Изображение с импульсной помехой



Изображение после применения АМФ

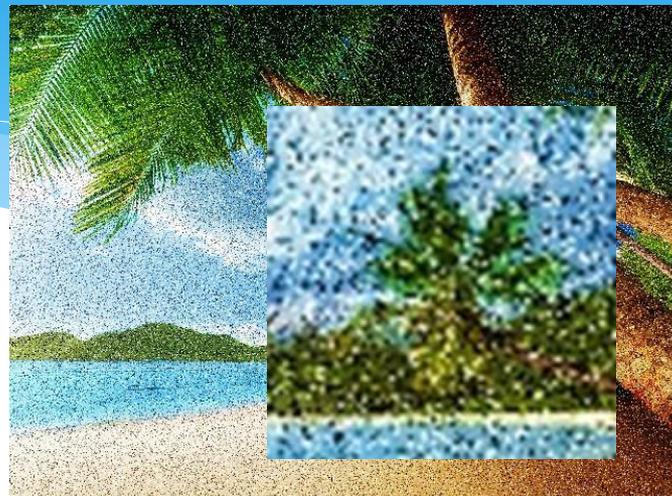


Изображение после применения АМФ-МП

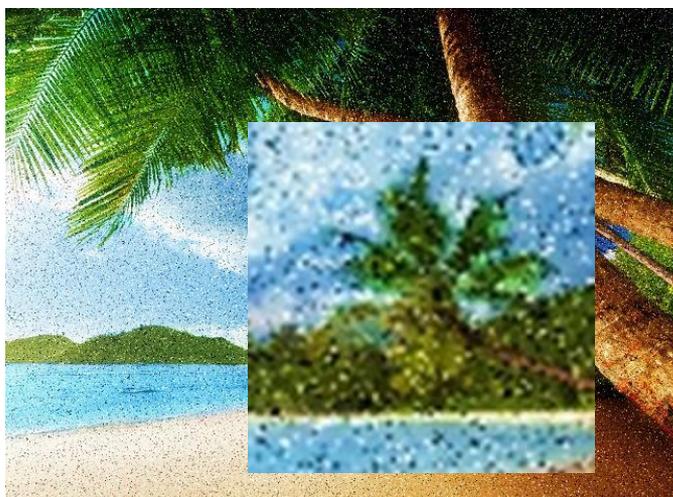
# В чём же отличие?



Исходное изображение



Изображение с импульсной помехой



Изображение после применения АМФ



Изображение после применения АМФ-МП

# Оценка разработанного алгоритма

- + Значительное расширение возможностей АМФ.
- + Снижение излишнего вмешательства со стороны пользователя.
- Экспериментально полученное пороговое значение может уточняться для отдельных категорий изображений при ранжировании их по различным параметрам, например по средней яркости, по оценкам контрастности, по распределению цветов, что требует дальнейших исследований.