



САНКТ-ПЕТЕРБУРГСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ПЕТРА ВЕЛИКОГО
ИНСТИТУТ КОМПЬЮТЕРНЫХ НАУК И ТЕХНОЛОГИЙ

НЕДЕЛЯ НАУКИ 2019

МАТЕМАТИЧЕСКИЕ АСПЕКТЫ МАСШТАБИРОВАНИЯ ИЗОБРАЖЕНИЯ С УЧЕТОМ СОДЕРЖИМОГО

Выполнили:

Лебедикова Валерия (гр.
3530203/80001)

Иванов Вячеслав (гр. 3530203/80001)

Научный руководитель:

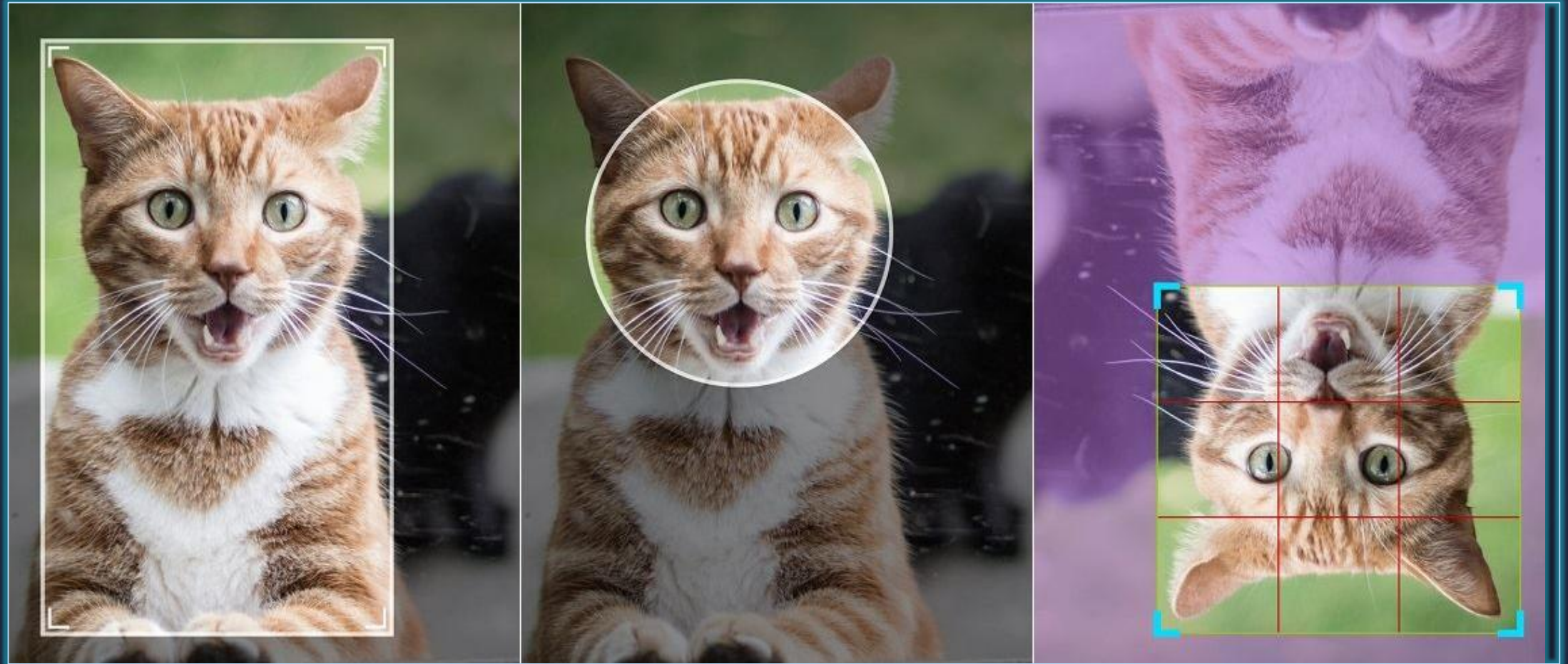
Доцент каф. «Высшая математика»

Филимоненкова Надежда Викторовна

Scaling



CROPPING



Seam carving

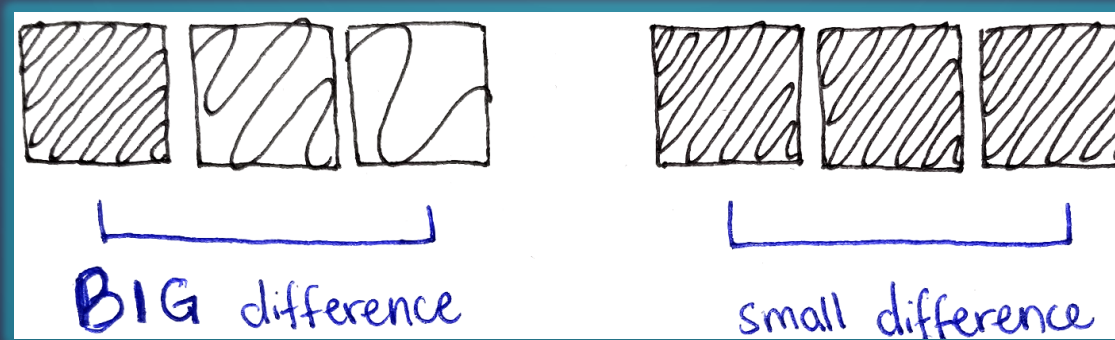


Задачи НИР

- Изучить алгоритм вырезания швов.
- Разобрать математические подходы к измерению “энергии”.
- Реализовать программу для рассматриваемого алгоритма с различными методами измерения энергии.
- Дополнение: реализовать метод вставки швов.

Алгоритм вырезания швов

- Назначаем каждому пикселю энергию в зависимости от того, насколько он неоднороден (отличается от окружающих его пикселей).

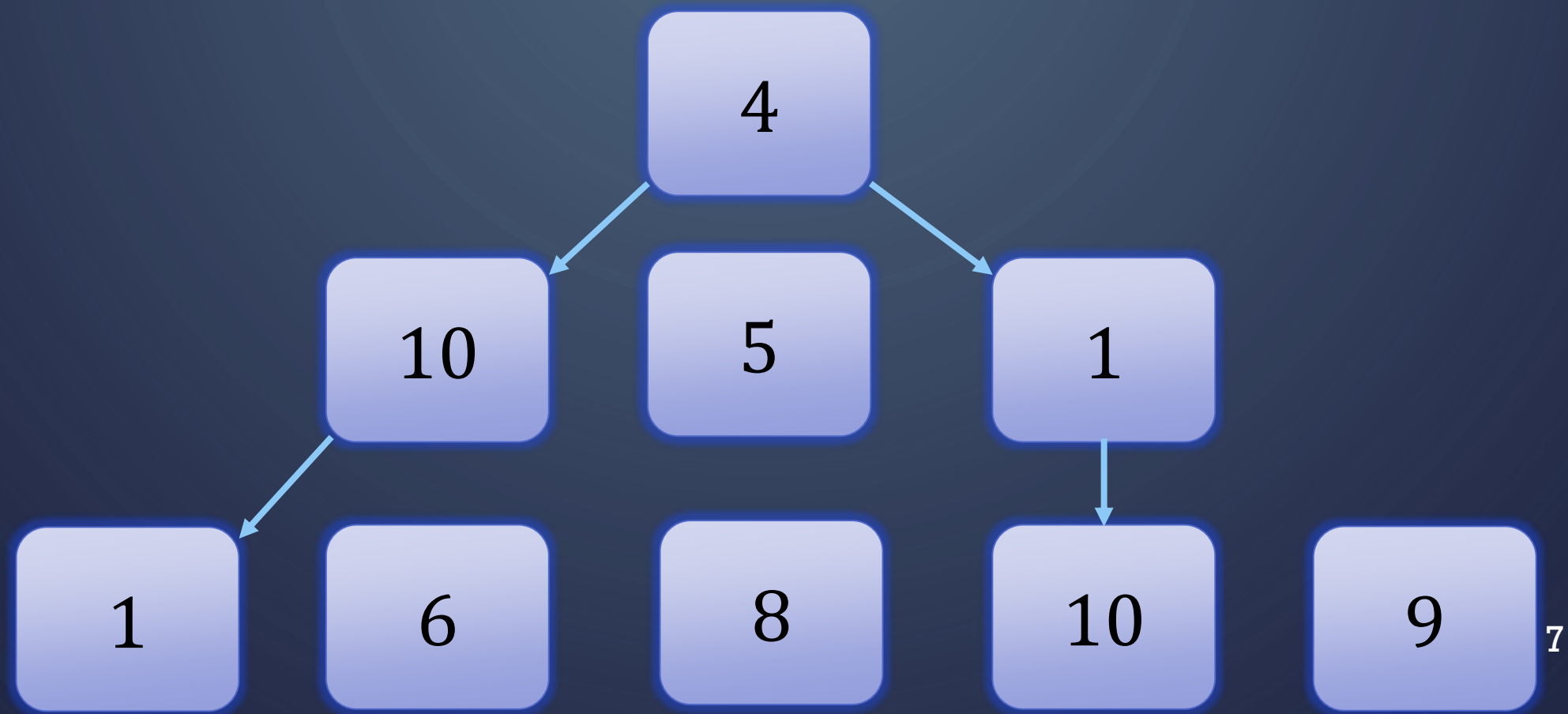


- Применяем динамическое программирование, чтобы найти путь через изображение с наименьшей энергией.
- Удаляем найденный путь.

Поиск низкоэнергетического шва

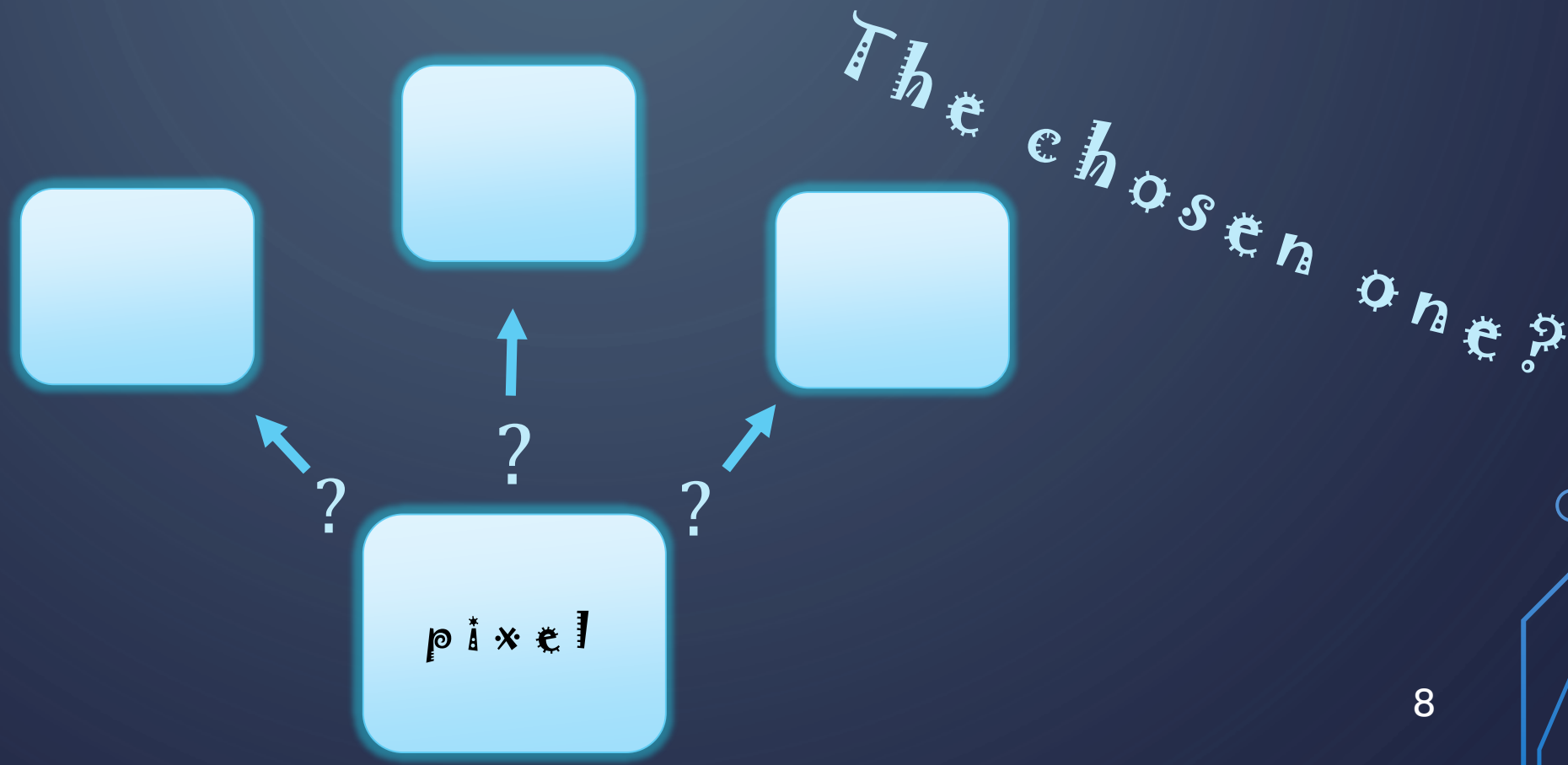
Динамическое программирование помогает разбить проблему на подзадачи.

Проблема в том, что при принятии решения о следующем шаге мы не учитываем часть шва впереди, ведь самый низкоэнергетический шов не обязательно проходит через пиксели с наименьшей энергией, например:



Поиск низкоэнергетического шва

Для каждого пикселя ряда изучаем три соседних пикселя в строке ниже. Фундаментальный выбор — какой из швов продолжить?



Поиск низкоэнергетического шва

Находим пиксель в последнем ряду с минимальной энергией. Он хранит в себе весь путь до него. Осталось подняться от последнего ряда к первому, используя жадный алгоритм.

1	4	3	5	2
3 + 1 4	2 + 1 3	5 + 3 8	2 + 2 4	3 + 2 5
5 + 3 8	1 + 3 4	4 + 3 7	2 + 4 6	1 + 4 5

Сумма
текущего
пикселя и
минимального
из соседей
верхнего ряда

Самый низкоэнергетический
шов

Convolution

Алгоритмы выделения границ обычно реализуются с помощью свёртки.

Свёртка — это операция вычисления нового значения выбранного пикселя в зависимости от соседних. Для этого нужна особая матрица - ядро.

Для работы с матрицами изображений применяется двумерная дискретная свёртка между данным изображением и ядром:

$$f(x, y) = K * I(x, y) = \sum_{i=-l}^l \sum_{j=-l}^l K(i, j) I(x - i, y - j)$$

Где $f(x, y)$ – обработанное изображение, $I(x, y)$ – оригинальное изображение, K – ядро свёртки, $*$ - операция свёртки

Image Convolution Example

Исходное
изображение

	x1	x2	x3	
	x4	x0	x5	
	x6	x7	x8	

Влияющие пиксели

Матрица
свёртки (ядро)

*

m1	m2	m3
m4	m0	m5
m6	m7	m8

=

Полученное
изображение

		y		

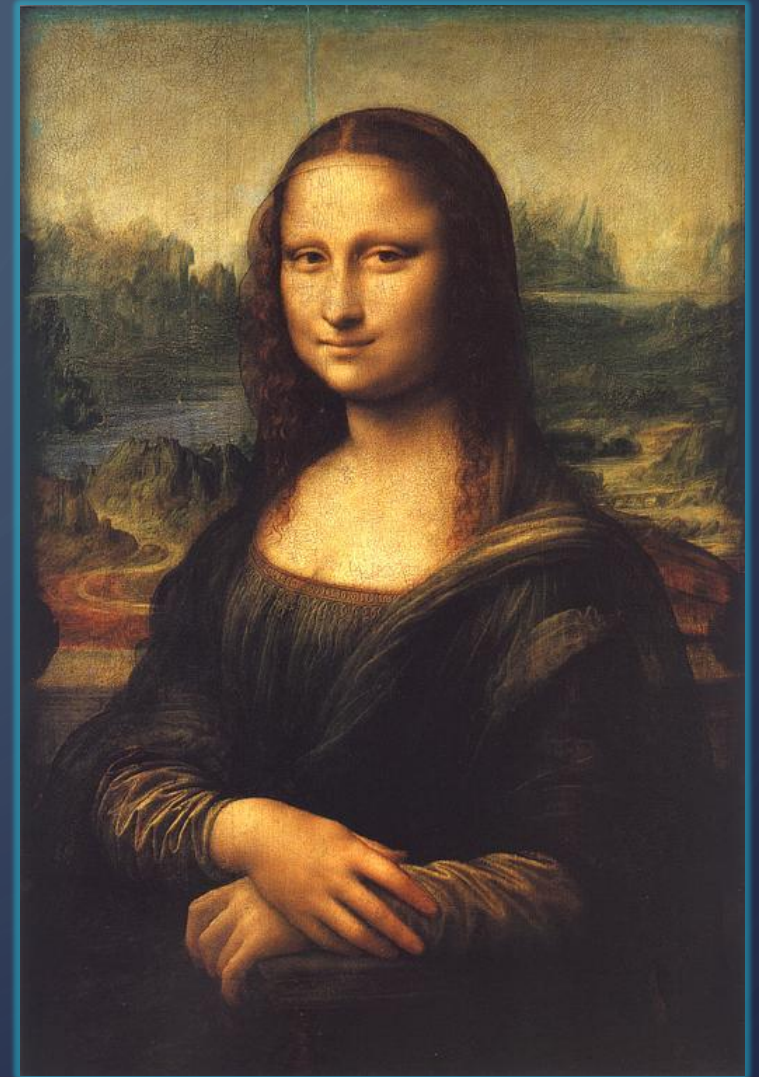
Зависимый пиксель

Edge detection

Методы выделения границ можно разделить на две категории: методы, основанные на поиске максимумов, и методы, основанные на поиске нулей.



Original Images



Gradient based Edge operator

Вектор градиента изображения $I(x, y)$:

$$\nabla I(x, y) = \begin{bmatrix} \frac{\partial I(x, y)}{\partial x} \\ \frac{\partial I(x, y)}{\partial y} \end{bmatrix} = \begin{bmatrix} I_x \\ I_y \end{bmatrix}$$

Евклидова норма градиента:

$$|\nabla I(x, y)| = \sqrt{I_x^2 + I_y^2}$$

Norms

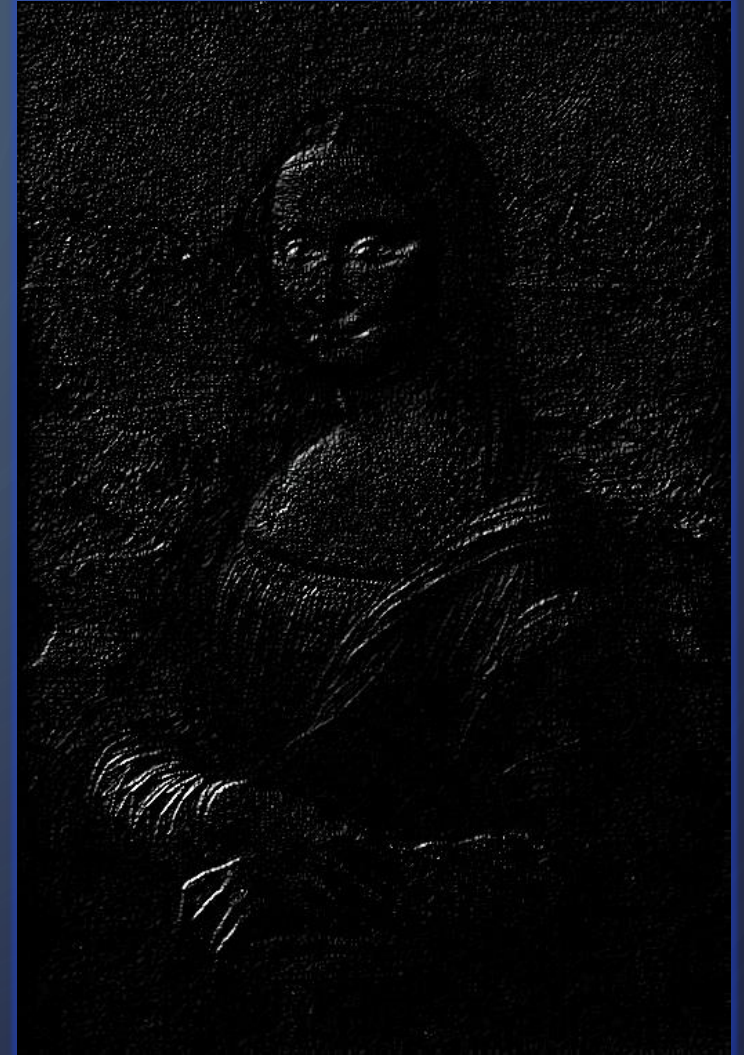


Image Derivative

Определение производной первого порядка:

$$\frac{dI(x)}{dx} = \lim_{\Delta x \rightarrow 0} \frac{I(x + \Delta x) - I(x)}{\Delta x}$$

Так как минимальная разница Δx между пикселями на изображении равна 1, то:

$$\frac{dI(x)}{dx} \cong \frac{I(x + 1) - I(x)}{1} = I(x + 1) - I(x)$$

Вычисление производной между пикселями, окружающими данный эквивалентна разности между ними:

$$\frac{dI(x)}{dx} \cong I(x + 1) - I(x - 1)$$

Sobel Operator

Оператор Собеля использует данные ядра, с которыми сворачивают исходное изображение для вычисления приближённых значений производных по горизонтали и вертикали.

$$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

Matrices

Convolution

$$\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

Sobel Operator Example

$$\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$



$$\begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix}$$

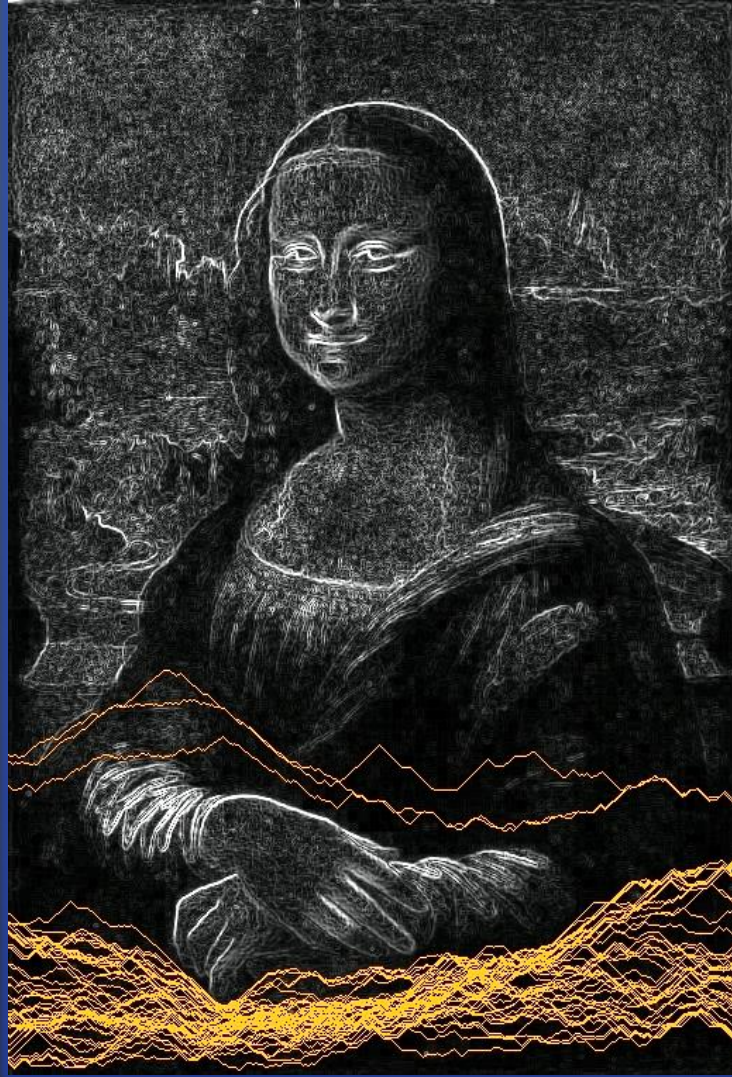
*

Previous pixel	Current x0	Next pixel
-------------------	---------------	---------------

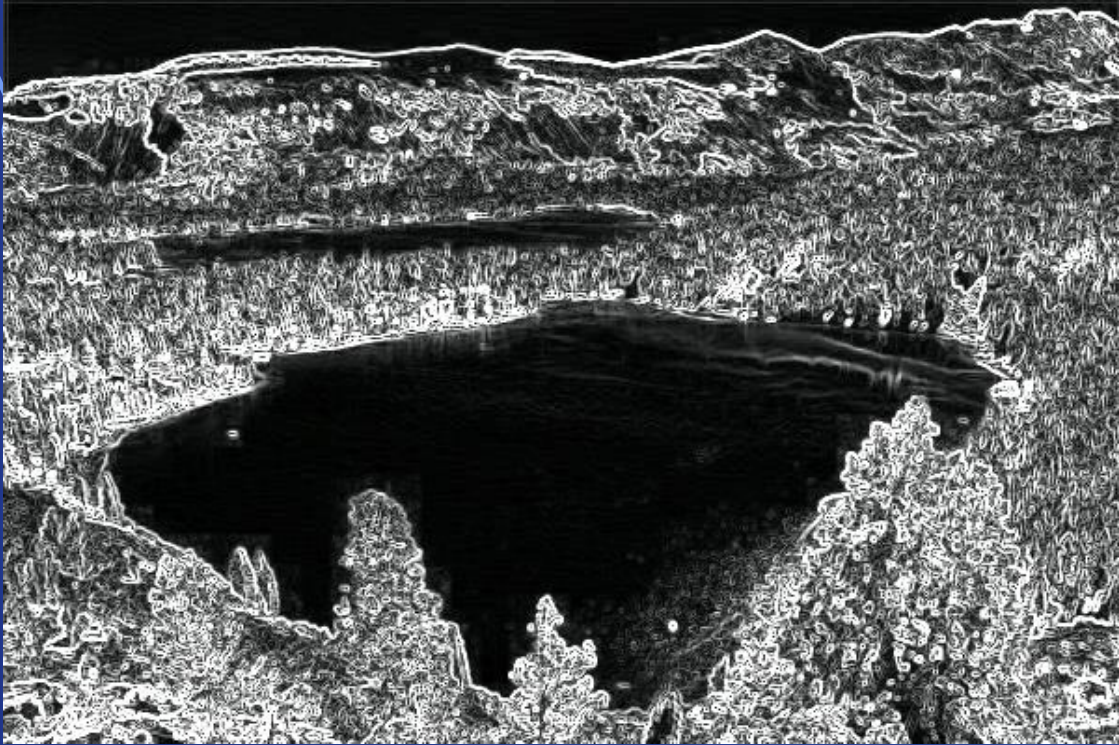
= Next pixel - previous pixel

⇒ Обыкновенное вычитание в
изображениях это аппроксимация взятия
производной

Our results



Our results



Prewitt Operator

Алгоритм оператора Прюитта подобен алгоритму оператору Собеля, за исключением использования другой матрицы:

$$\begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$$

Matrices

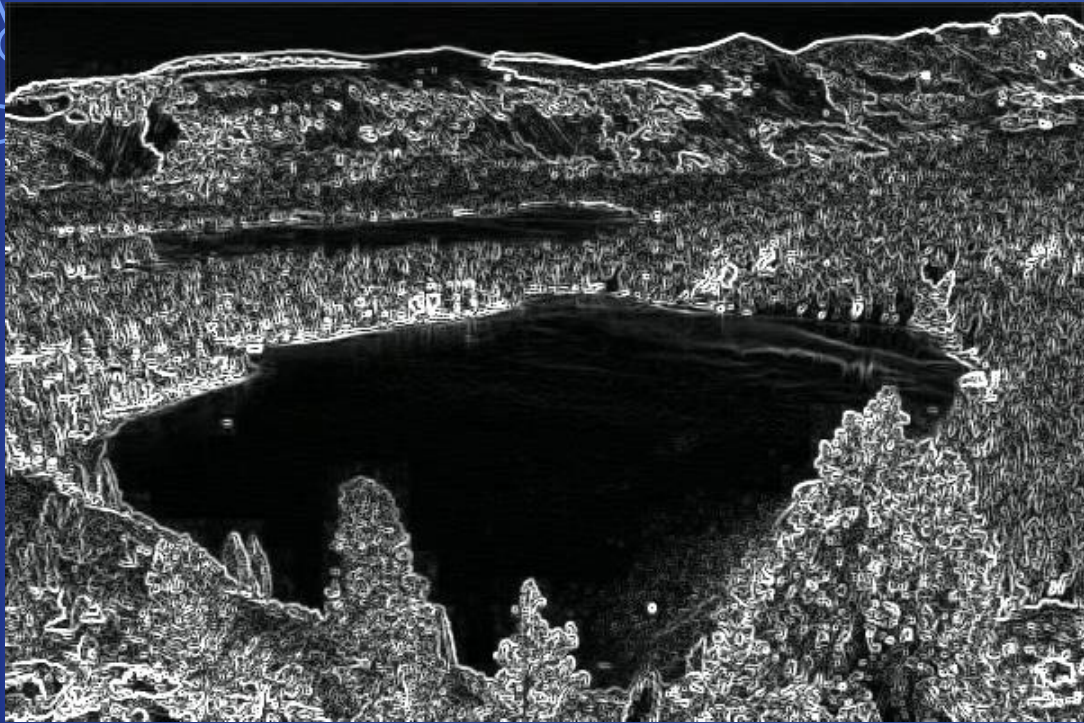
Convolution

$$\begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

Our results



Our results



Laplacian based Edge operator

Данный оператор основан на Лапласиане изображения.
Пиксель, в котором вторая производная изображения
(разность разности) становится равной нулю,
помечается как граничный.

$$\nabla^2 I(x, y) = \frac{\partial^2}{\partial x^2} I(x, y) + \frac{\partial^2}{\partial y^2} I(x, y)$$

Marr-Hildreth (Laplacian of Gaussian) Operator

Данный метод является улучшенной версией оператора Лапласа. Он использует дополнительно ядро сглаживания Гаусса. С помощью него эффективно устраняются шумы.

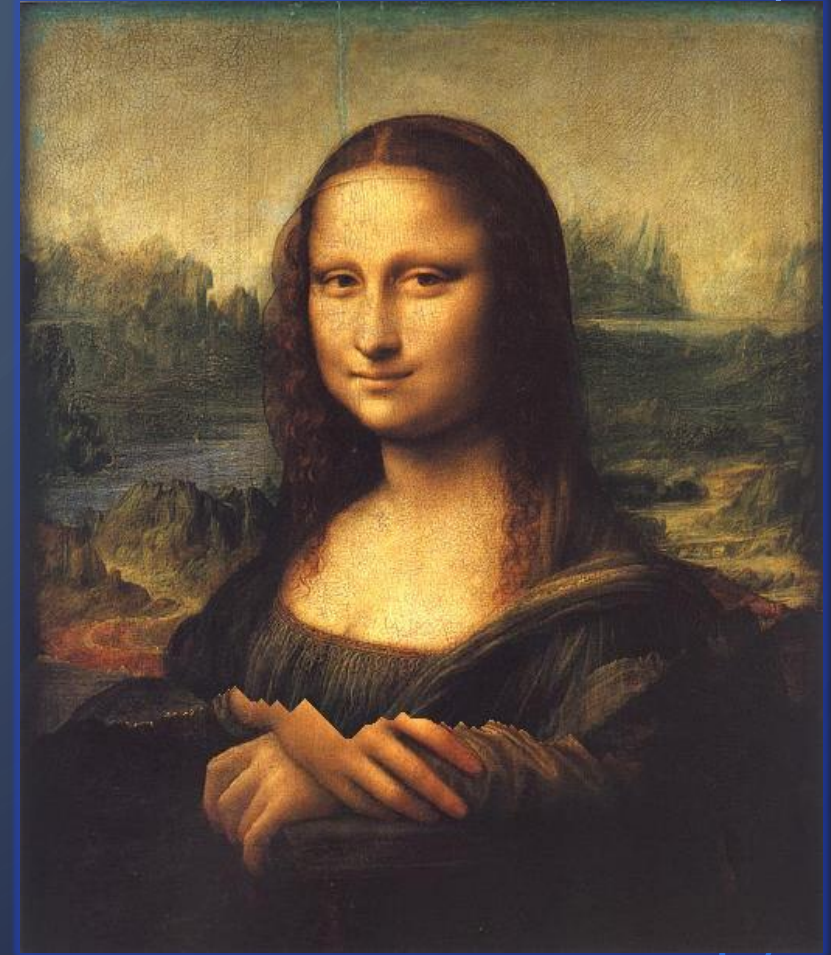
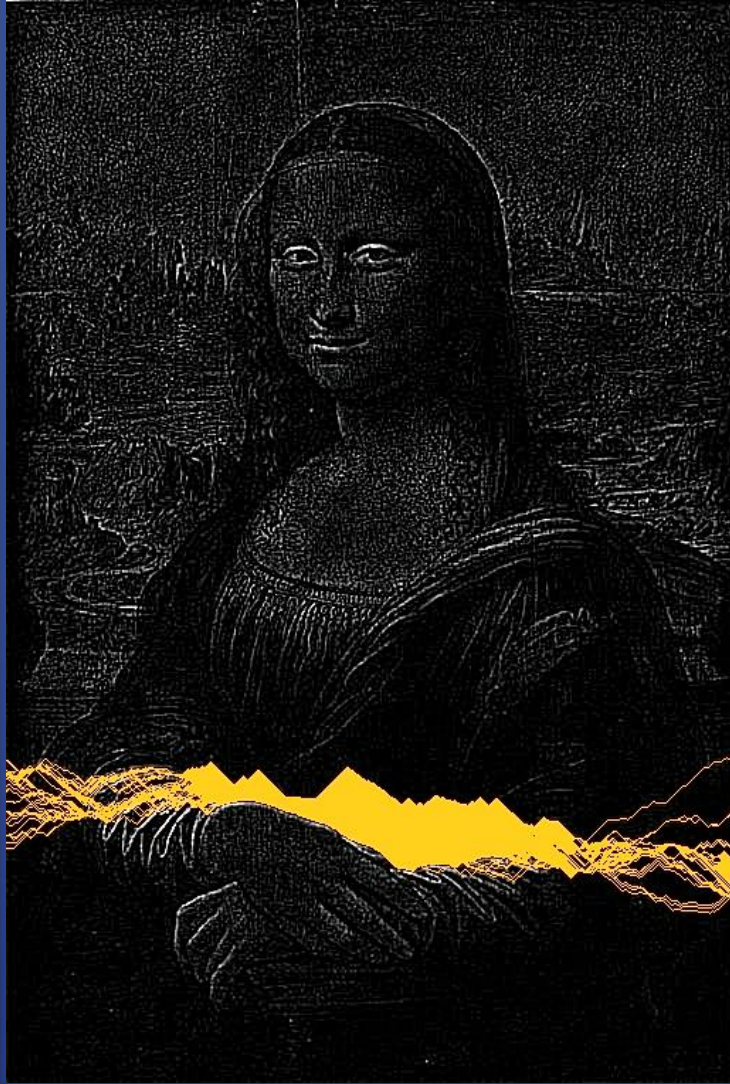
Где g — ядро Гаусса

$$\nabla^2(I * g) = I * \nabla^2 g$$

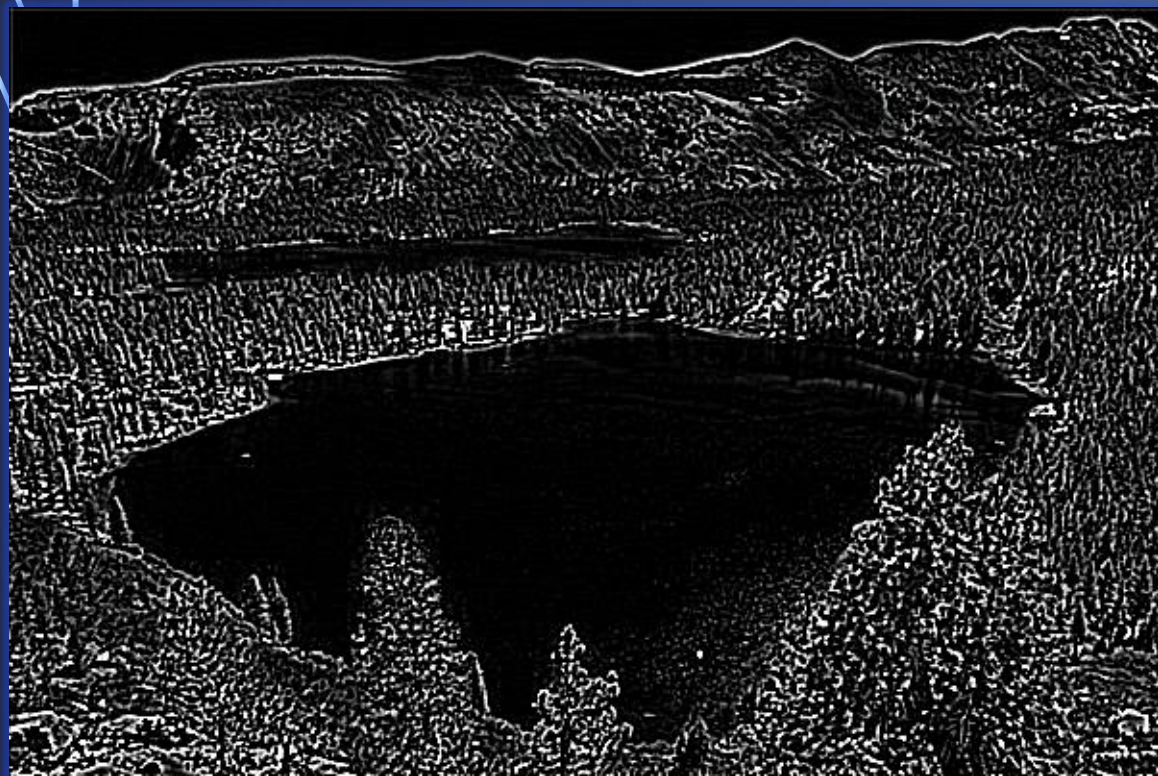
$$I * \nabla^2 g = I * \left(\frac{\partial^2}{\partial x^2} g + \frac{\partial^2}{\partial y^2} g \right) = I * \frac{\partial^2}{\partial x^2} g + I * \frac{\partial^2}{\partial y^2} g$$

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & 4 & 1 \\ 0 & 1 & 0 \end{bmatrix} \text{ Gaussian} \quad \text{Laplacian} \quad \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

Our results



Our results



Entropy edge detection

Энтропия – это то, сколько информации неизвестно о системе
(характеризует степень неопределенности)

Энтропия Шеннона:

$$S(x) = - \sum_{i=1}^n p_i \cdot \log_2(p_i)$$

Энтропия Реньи (обобщение энтропии Шеннона):

Пусть $p_i = p_1, \dots, p_k$ – распределение вероятности (гистограмма)
для изображения с $k = 256$ оттенками серого, где 0 – черный, 255 – белый;
остальные значения – оттенки серого

$$H_{\alpha}(P) = \frac{1}{1 - \alpha} \cdot \log_2 \sum_{i=0}^{255} (p_i)^{\alpha}$$

Entropy edge detection

Цель: получить бинарное изображение, на котором можно будет выделить границы между черным и белым - швы

t – это порог яркости, при котором пиксели разделяются на принадлежность объекту А или В
 t^* – оптимальное граничное значение, при котором $P_B = P_A = 0,5$

$$P_A = \sum_{i=0} p_i - \text{вероятность встретить пиксель } i \text{ из класса } A$$

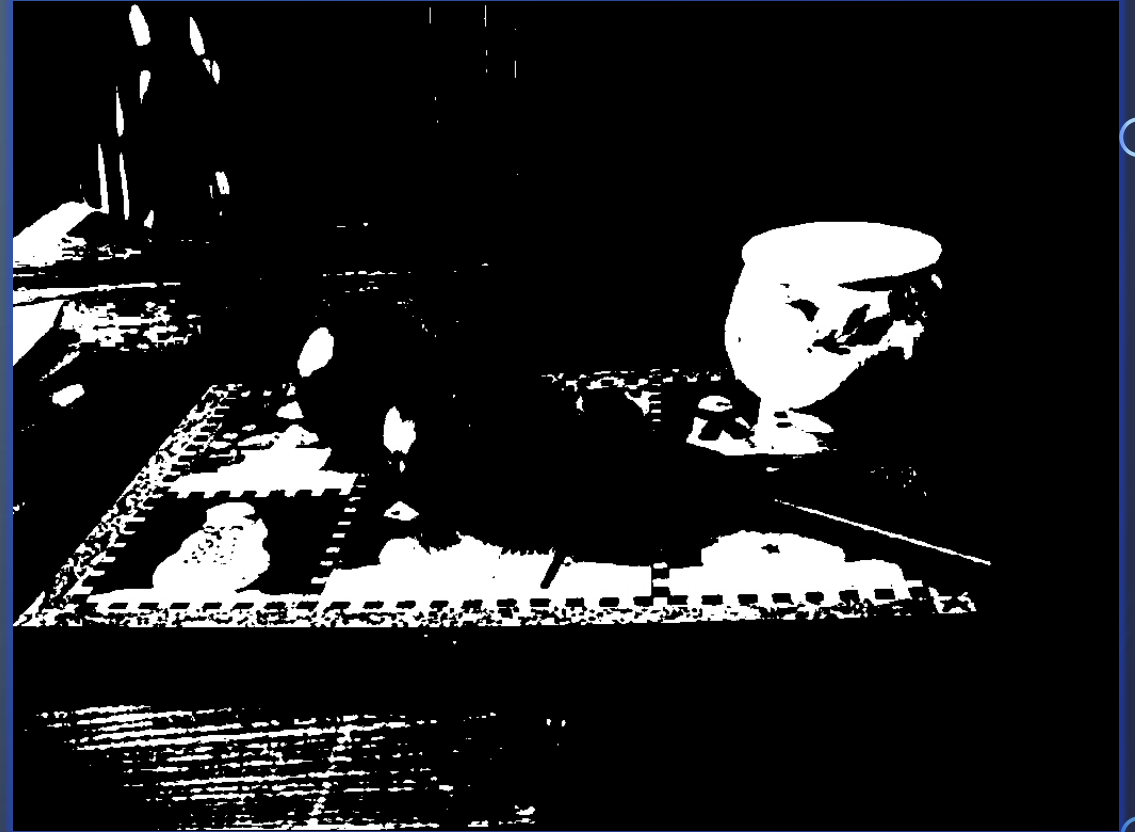
$$P_B = \sum_{i=t+1}^{255} p_i = 1 - P_A$$

Энтропия порядка α для каждого класса:

$$H_{\alpha}^A(t) = \frac{1}{1-\alpha} \cdot \log_2 \sum_{i=0}^t (p_i/P_A)^{\alpha}$$

$$H_{\alpha}^B(t) = \frac{1}{1-\alpha} \cdot \log_2 \sum_{i=t+1}^{255} (p_i/P_B)^{\alpha}$$

Binary Image Result



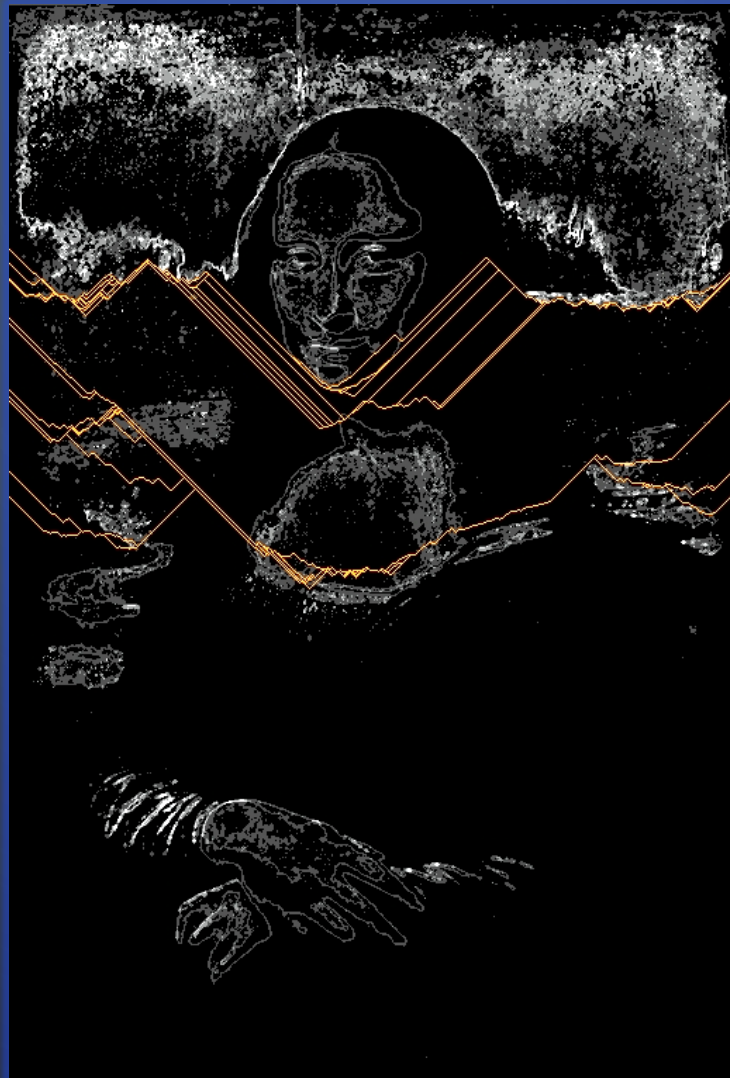
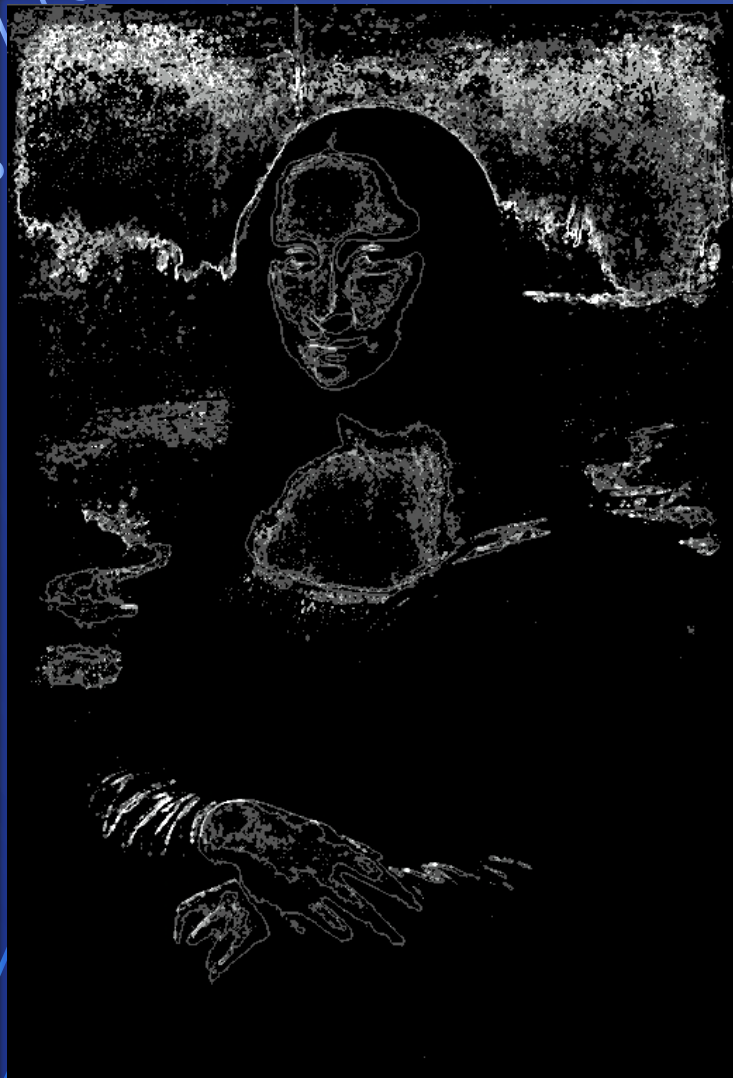
Entropy edge detection

Special Convolution

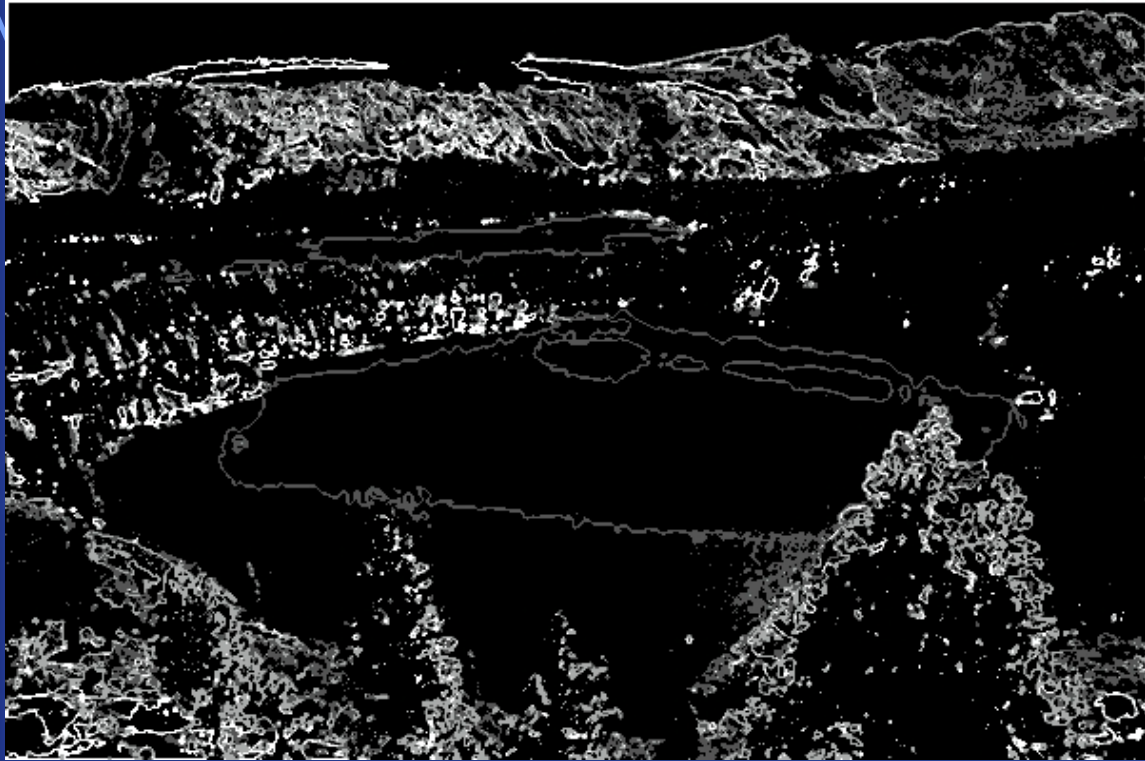
$$\begin{bmatrix} I(x-1, y-1) & I(x, y-1) & I(x+1, y-1) \\ I(x-1, y) & I(x, y) & I(x+1, y) \\ I(x-1, y+1) & I(x, y+1) & I(x+1, y+1) \end{bmatrix}$$

Matrix

Our results



Our results



Seam Insertion

$$newPx(x, y) = Avg(px(x, y - 1), px(x, y + 1))$$





Список литературы

- ❖ Avidan Sh., Shamir Ar., *Seam Carving for Content-Aware Image Resizing*. – 2007. – Режим доступа: <https://perso.crans.org/frenoy/matlab2012/seamcarving.pdf>
- ❖ Shubhashree Savant, *A Review on Edge Detection Techniques for Image Segmentation*. – Август 2014. – Режим доступа: https://www.researchgate.net/publication/291975278_A_Review_on_Edge_Detection_Techniques_for_Image_Segmentation
- ❖ Mohamed A. El-Sa, Muhammad Atta Othman Ahmed, *Using Renyi's entropy for edge detection in level images*. – Январь 2011. – Режим доступа: https://www.researchgate.net/publication/322939200_Using_renyi's_entropy_for_edge_detection_in_level_images
- ❖ *Динамическое программирование в реальном мире: вырезание швов*. - Июнь 28, 2019. – Режим доступа: <https://habr.com/en/post/458110/?mobile=no>
- ❖ Rashmi, Mukesh Kumar, and Rohini Saxena, *ALGORITHM AND TECHNIQUE ON VARIOUSEDGE DETECTION: A SURVEY*. – Июнь 2013. – Режим доступа: <http://aircconline.com/sipij/V4N3/4313sipij06.pdf>
- ❖ Raman Maini & Dr. Himanshu Aggarwal, *Study and Comparison of Various Image Edge Detection Techniques*. – Март 2009. – Режим доступа: <https://pdfs.semanticscholar.org/958e/a430ebece9b14671428cd030a68a6ba612b8.pdf>

The image features a dark blue background with a subtle radial gradient. In the four corners, there are decorative white line art elements resembling circuit boards or neural network connections, with lines and small circles extending from the edges towards the center.

Thank you
for your
attention