БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Кафедра интеллектуальных информационных технологий

ОТЧЁТ  
по лабораторной работе №2  
по дисциплине

«ОБРАБОТКА ИЗОБРАЖЕНИЙ В ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМАХ»

Выполнил студент гр. 321701

Романов К.В.

Проверил

Сальников Д. А.

Минск 2025

**Тема:** Предварительная обработка изображений.

**Задание:**

Реализовать один из фильтров на любом языке программирования или же используя сторонние библиотеки реализовать программу, которая может применить различные фильтры к исходному изображению.

**Листинг кода:**

from PIL import Image, ImageFilter, ImageOps

import matplotlib.pyplot as plt

def apply\_filters(image\_path):

# Открываем исходное изображение

image = Image.open(image\_path)

# Применяем фильтры

filters = {

"Оригинал": image,

"Размытие": image.filter(ImageFilter.BLUR),

"Контур": image.filter(ImageFilter.CONTOUR),

"Резкость": image.filter(ImageFilter.SHARPEN),

"Оттенки серого": ImageOps.grayscale(image),

"Усиление краёв": image.filter(ImageFilter.EDGE\_ENHANCE)

}

# Сохраняем фильтрованные изображения

for name, img in filters.items():

filename = f"filtered\_{name.replace(' ', '\_').lower()}.jpg"

img.save(filename)

# Отображаем все фильтры через matplotlib

plt.figure(figsize=(12, 8))

for i, (name, img) in enumerate(filters.items(), 1):

plt.subplot(2, 3, i)

plt.imshow(img if img.mode == 'RGB' else img, cmap='gray')

plt.title(name)

plt.axis('off')

plt.tight\_layout()

plt.show()

if \_\_name\_\_ == "\_\_main\_\_":

apply\_filters("/home/kirillromanoff/University/OIIS/2/1.png")

Пример результата работы программы:

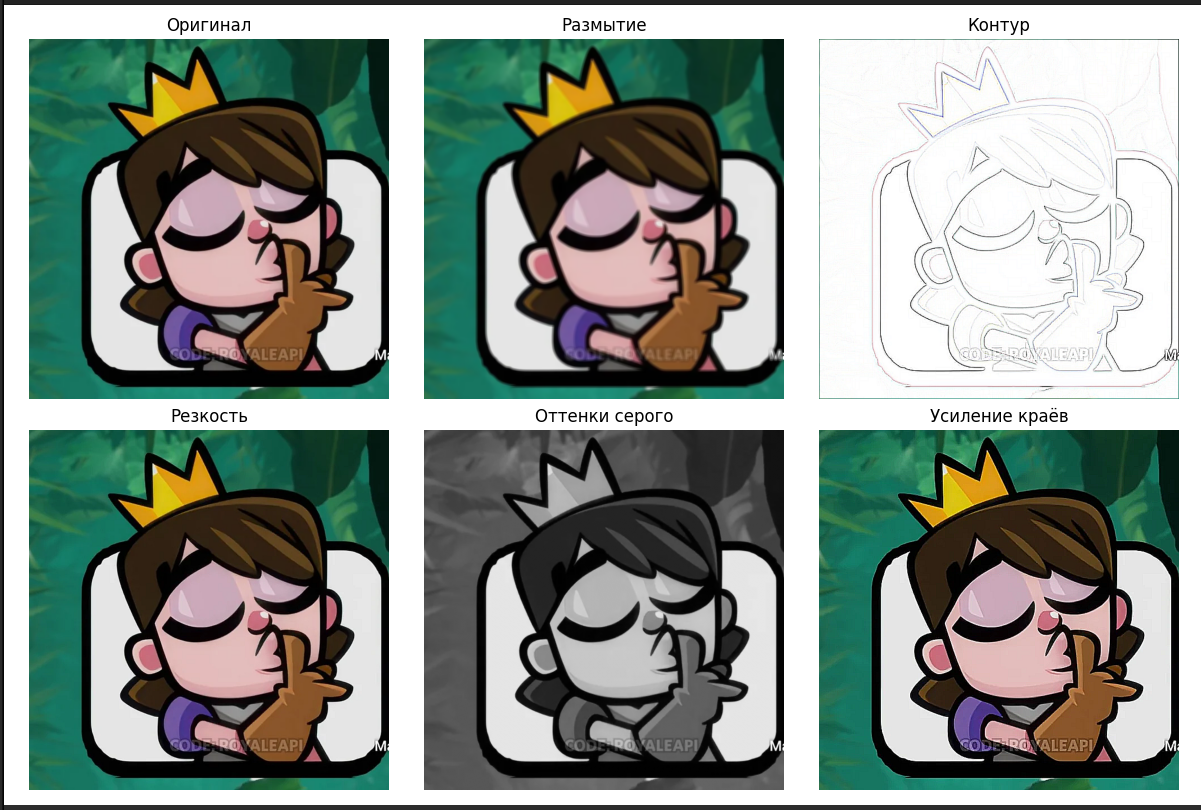


Рис.1 - Применение фильтра повышения резкости на исходное изображение

**Вывод:** в рамках лабораторной работы была реализована программа, позволяющая применить фильтры гауссово размытие, медианный фильтр, фильтр Собеля, билатеральный фильтр, повышение резкости, адаптивная пороговая обработка, обнаружение границ, эффект акварели, сепия, увеличение контраста.