



Обработка изображений на спутниковых снимках с помощью нейронных сетей

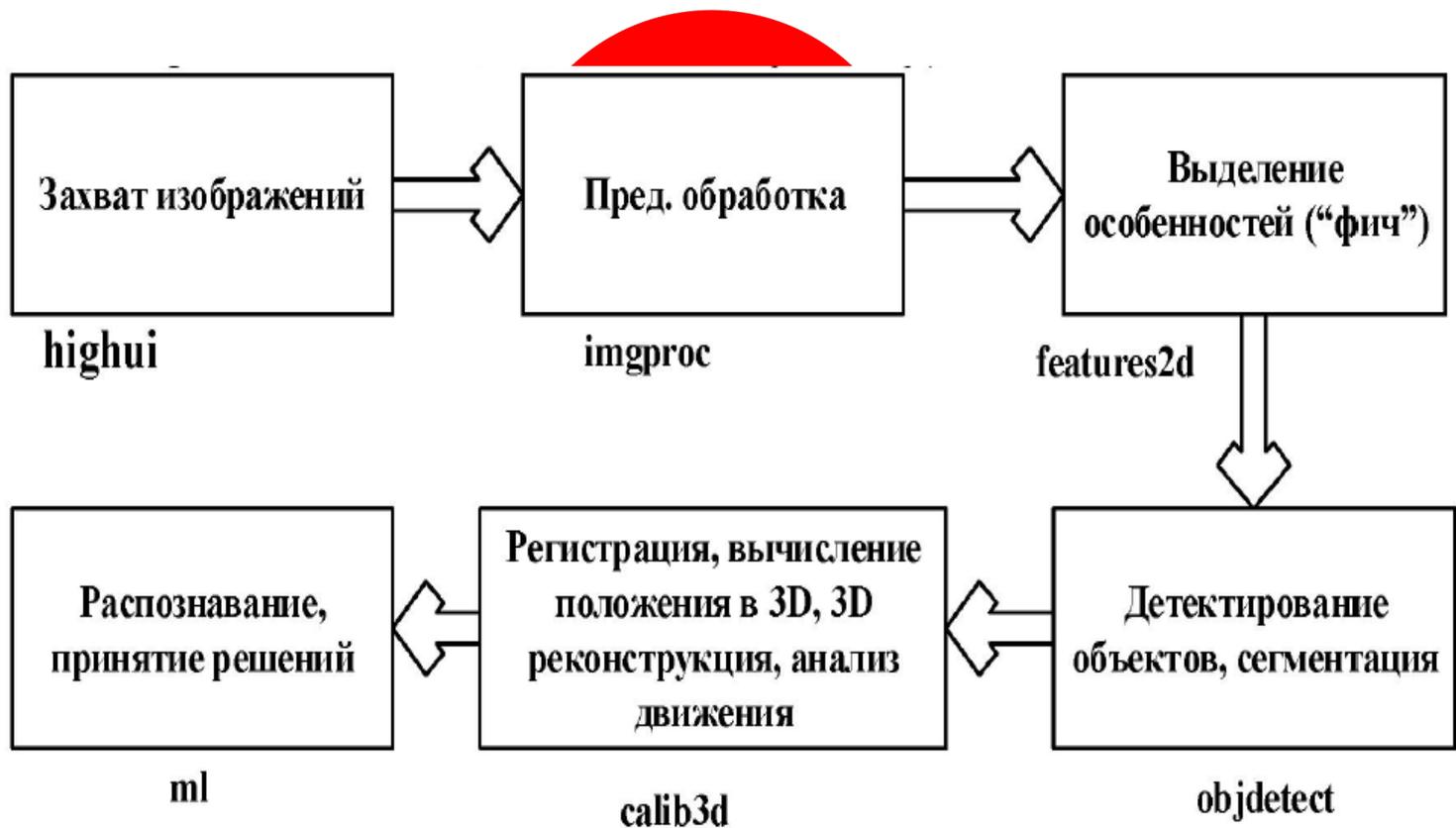
Касатиков Н.Н., Гомозов О.А. к.т.н., Толмачев С.А., Фадеева А.Д., Рогожин М.Е., к.т.н. Токарев А.В., к.т.н. Сытов А.О. ,
Макиеров М.И.

2021

АКТУАЛЬНОСТЬ



Рисунок 2.1 – Области применения систем распознавания образов и технического зрения



OpenCV

Рисунок 4.1 – Общая схема типового приложения с реализацией компьютерного зрения в пакете OpenCV

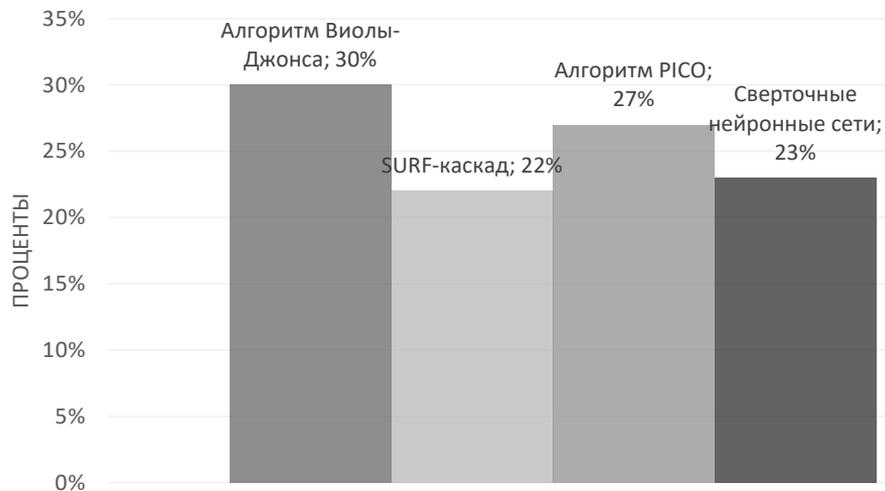


Рисунок 5.1 – Величина ошибки первого рода

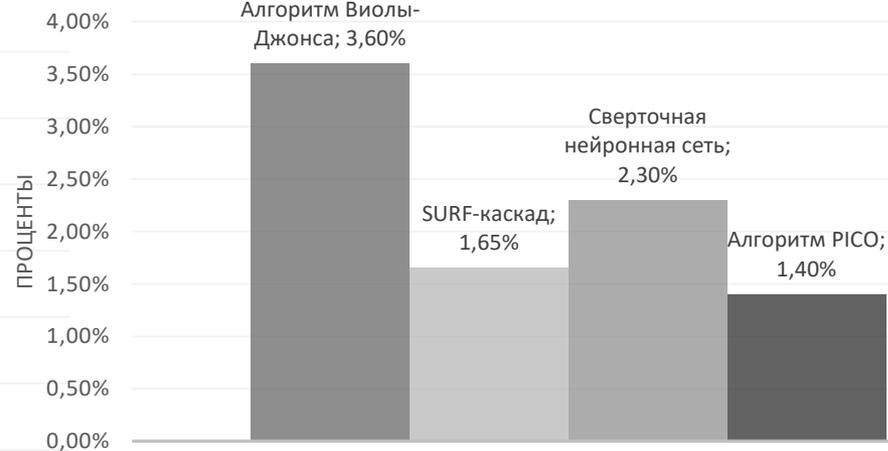


Рисунок 5.2 – Величина ошибки второго рода

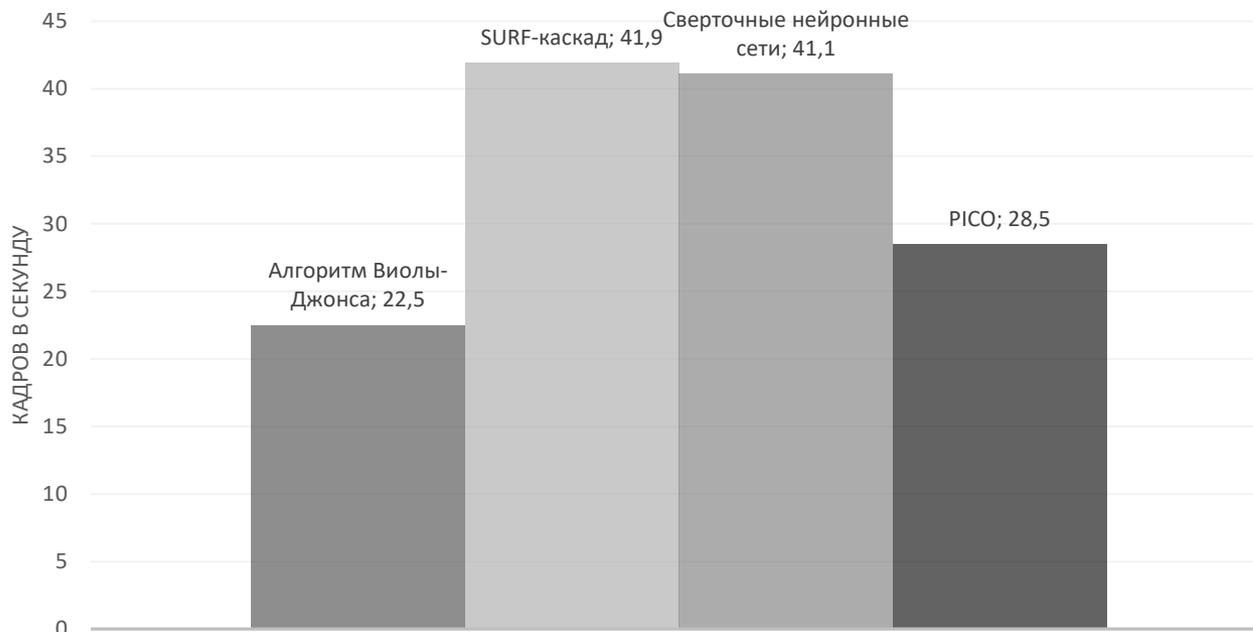
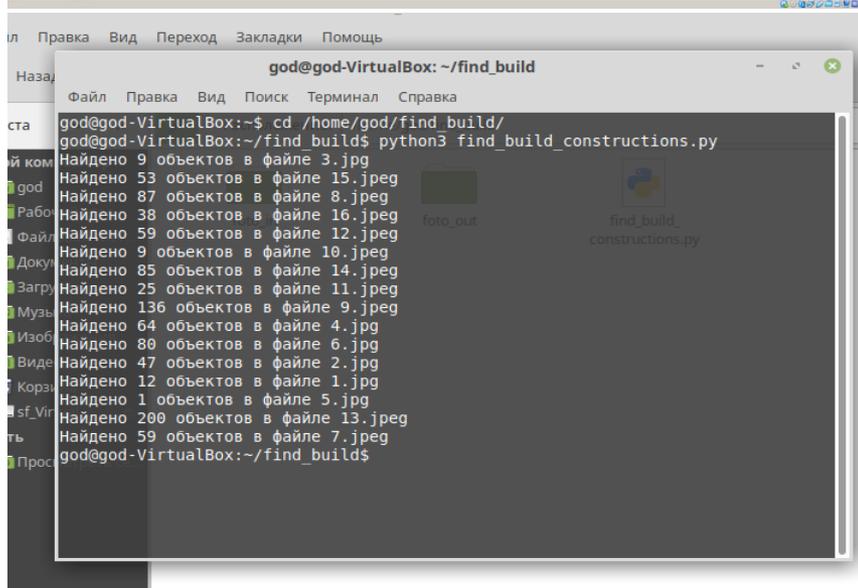
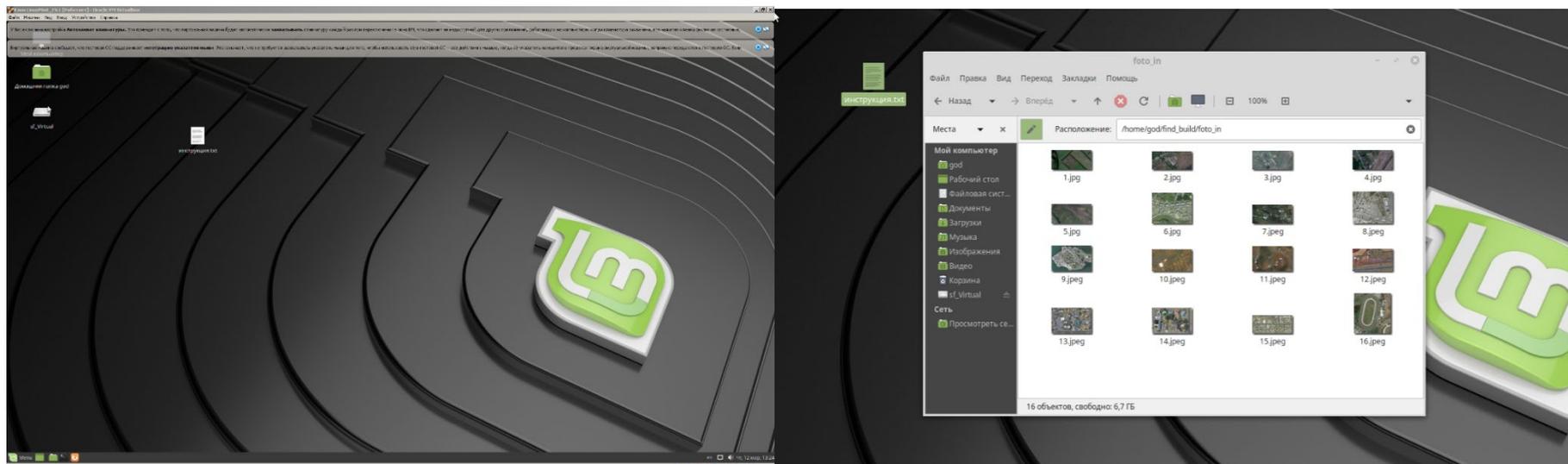


Рисунок 5.3 – Скорость обработки изображений

Научно-технические предложения по практической реализации алгоритмов распознавания в интересах распознавания зданий



4. Поиск контуров в изображении и подсчет количества зданий

Листинг 4

```
cnts = cv2.findContours(closed.copy(),
cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_SIMPLE)[1]
total = 0
# цикл по контурам
for c in cnts:
    rect = cv2.minAreaRect(c) # пытаемся вписать
прямоугольник
    box = cv2.boxPoints(rect) # поиск четырех вершин
прямоугольника
    box = np.int0(box) # округление координат
    cv2.drawContours(image, [box], -1, (0, 255, 0), 3)
    total += 1
    print("Найдено {0} объектов в файле {1}"
.format(total, file_name))
    cv2.imwrite(output_dir + '/' + file_name +
"_output.jpg", image)...
```

Рисунок 7.1 – Рабочие окна программы “Здание Y”

Тестирование программы “Здание Y”

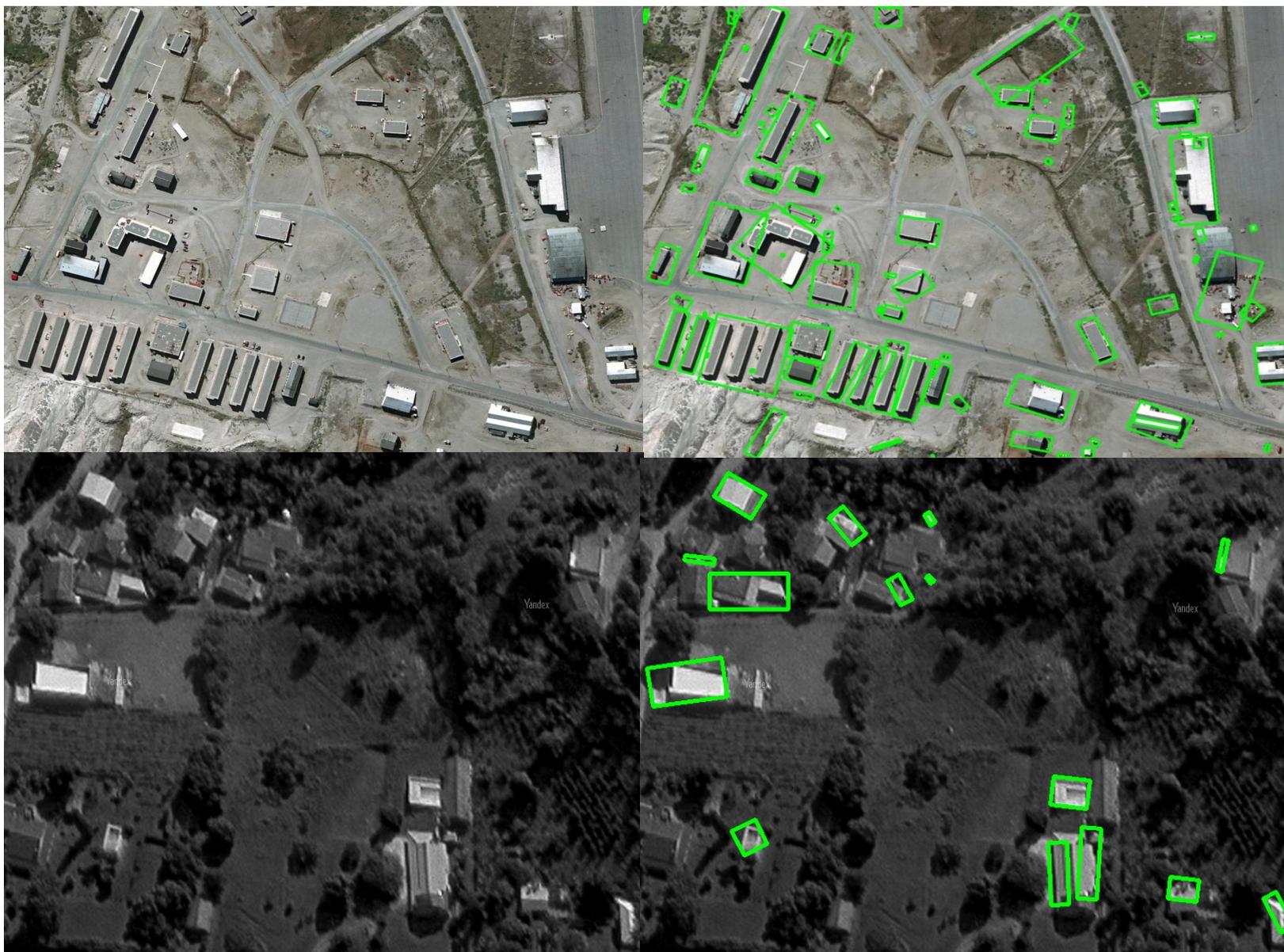


Рисунок 8.1 – Тестирование распознавания зданий

Тестирование программы “Дорога Z”

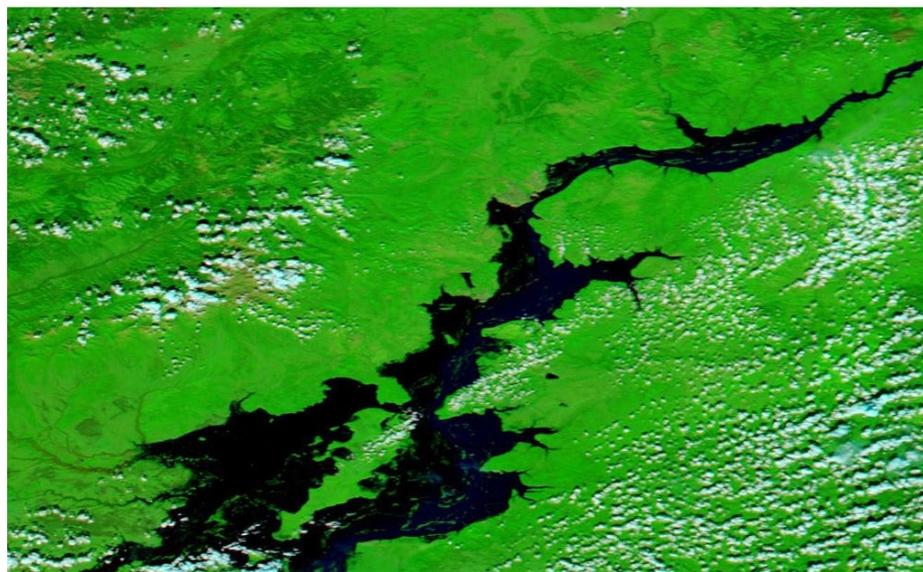
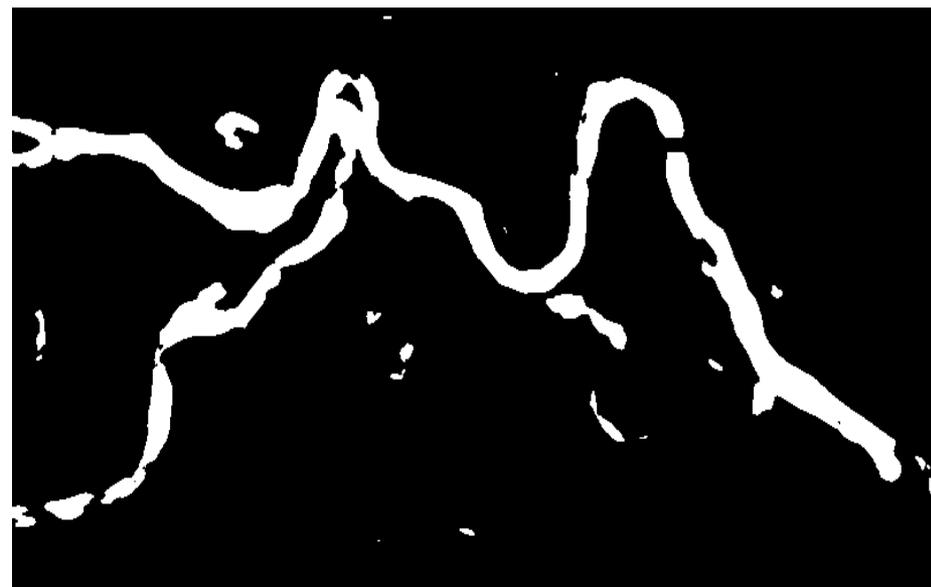


Рисунок 19.1 – Модернизация программы распознавания дорог

Научно-технические предложения по практической реализации алгоритмов распознавания в интересах распознавания дорог

```
C:\Windows\system32\cmd.exe - "python/python.exe" main.py
Microsoft Windows [Version 6.1.7601]
(c) Корпорация Майкрософт (Microsoft Corp.), 2009. Все права защищены.
C:\Users\Николай>cd C:\putsp\road_detection
C:\putsp\road_detection>"python/python.exe" main.py
Using TensorFlow backend.
2020-04-19 14:22:29.111649: I C:\Users\User\Source\Repos\tensorflow\tensorflow\
ore\platform\cpu_feature_guard.cc:140] Your CPU supports instructions that this
TensorFlow binary was not compiled to use: AVX
start
Введите путь к изображению: _
```

5. Обучение нейронной сети

Листинг 5

```
x = [] # Массив для входных изображений
y = [] # Массив для выходных изображений
original = ut.loadAllImages('dataset/input') # Входные
изображения
mask = ut.loadAllImages('dataset/output', bw=True) #
Выходные изображения
for o, m in zip(original, mask):
    x += ut.split_image(o, step_x=150, step_y=150)#
Разбиваем каждое изображение на небольшие части и добавляем
в тренировочные данные
    y += ut.split_image(m, step_y=150, step_x=150)#
Разбиваем каждое изображение на небольшие части и добавляем
в тренировочные данные
```



Рисунок 9.1 – Рабочие окна программы “Дорога Z”

Тестирование программы “Дорога Z”



Рисунок 10.1 – Тестирование распознавания дорог

Распознавание видео

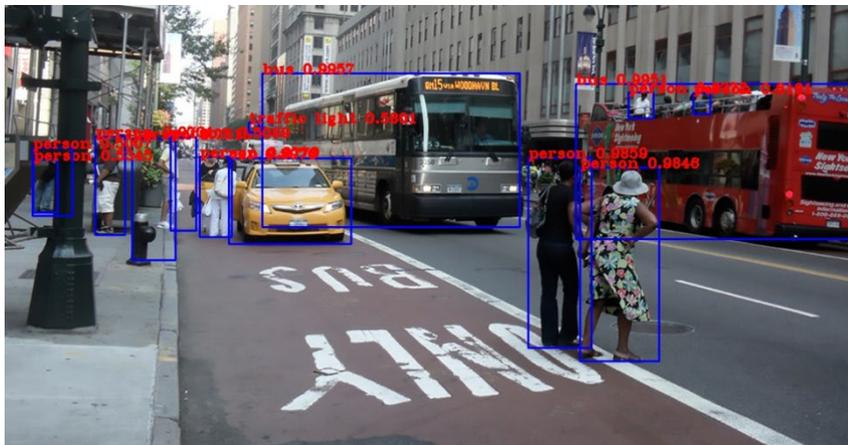


Рисунок 11. Тестирование распознавания MP4

Распознавание в ночное время

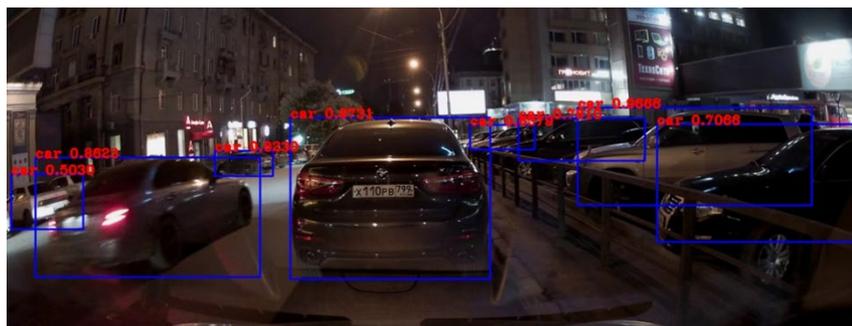


Рисунок 11.1. Тестирование распознавания в темное время

Анализ результатов, полученных в ходе проведенного в магистерской диссертации исследования

	Программа “ЗданиеУ”	Программа "Здание2"	Программа "Здание3"
Точность распознавания	90	88	85
Качество распознавания	97	91	97
Быстродействие	80	82	40

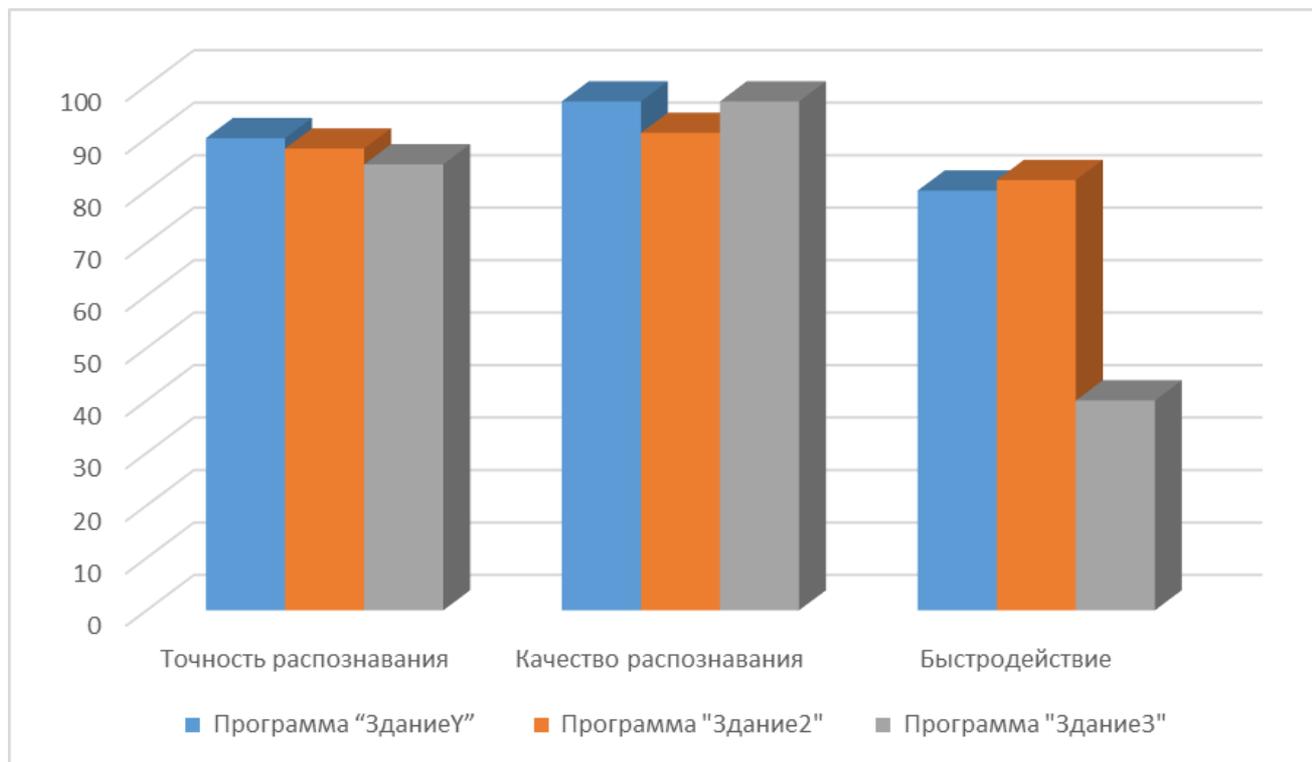


Рисунок 13.1 – Комплексное сравнение экспериментальной программы “Здание У” с аналогами

Анализ результатов, полученных в ходе проведенного в магистерской диссертации исследования

	Программа “ДорогаZ”	Программа "Дорога2"	Программа "Дорога3"
Точность распознавания	87	80	95
Качество распознавания	90	81	97
Быстродействие	85	81	89

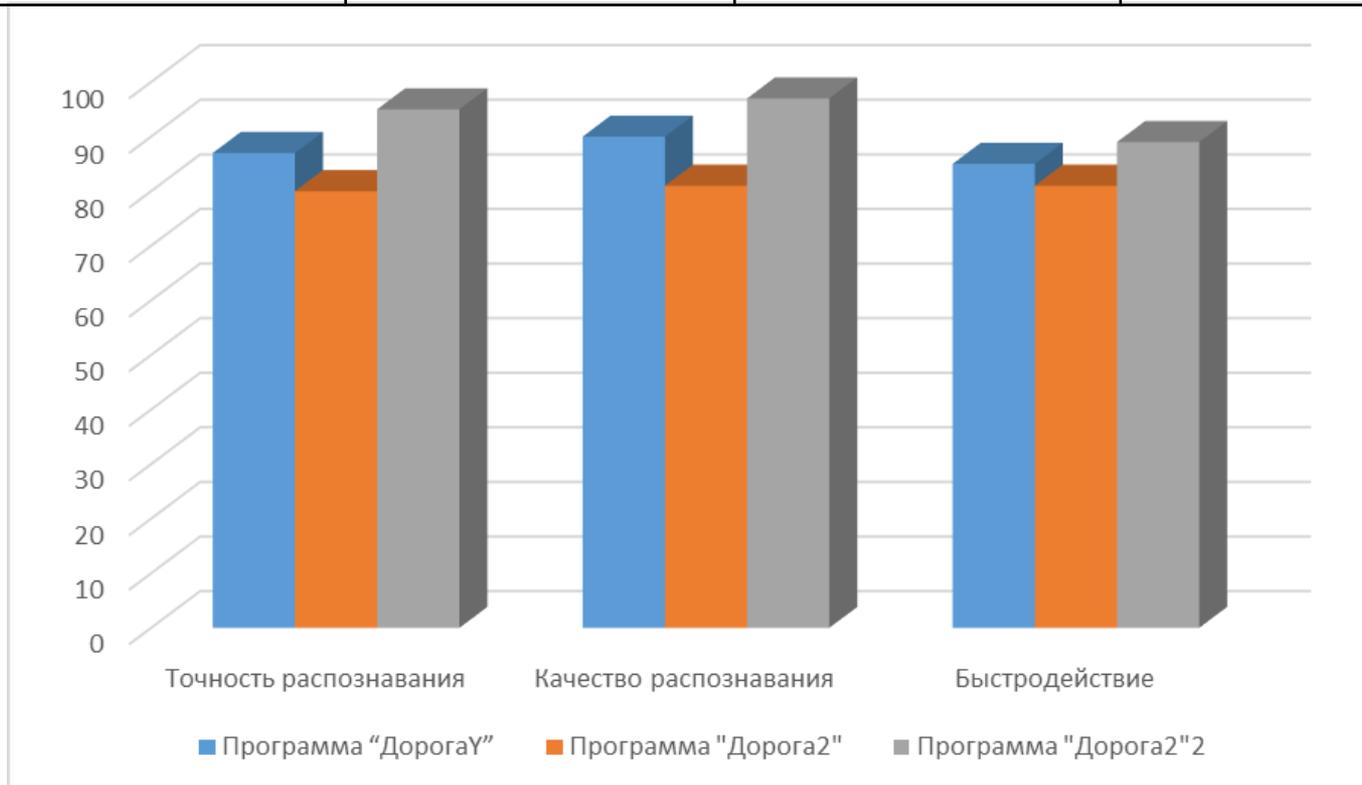


Рисунок 14.1 – Комплексное сравнение экспериментальной программы “Дорога Z” с аналогами

Выводы

- проведен теоретический анализ текущего состояния области распознавания топогеографических объектов
- представлен аналитический обзор доступных публикаций на тему исследования и результаты патентного поиска. Были найдены два самых интересных патента, а именно «Способ распознавания графических образов объектов» авторов Мингалев А.В., Агафонова Р.Р. и др, а так же «Способ обучения системы распознавания товаров на изображениях» авторов Бойко П.Ю., Виноградова М. С. и др на базе которых было принято решение отталкиваться в дальнейших разработках
 - представлен аналитический обзор доступных программных модулей распознавания объектов ,по результатам был выбран модуль OpenCV
 - проведено исследование методов и алгоритмов распознавания образов, сравнительный анализ, в результате чего была выбрана сверточная нейронная сеть, показав отличные результаты в скорости и точности
 - осуществлена разработка научно-технических предложений по практической реализации алгоритмов распознавания в интересах распознавания зданий и было решено воспользоваться библиотекой с открытым исходным кодом OpenCV для создания приложения
 - осуществлена разработка научно-технических предложений по практической реализации алгоритмов распознавания в интересах распознавания дорог и было решено воспользоваться сверточными нейронными сетями для создания приложения(и рек)
 - осуществлена разработка научно-технических предложений по практической реализации алгоритмов распознавания в интересах распознавания пиксельной разницы изображений было решено создать приложение с помощью языка C# из-за легкости в использовании
 - проведен анализ результатов, полученных в ходе проведенного в магистерской диссертации исследования ,а именно сравнение с образцами доступными на рынке и был выявлен ряд преимуществ перед бесплатными конкурентами , так же были обнаружены конкурентные плюсы , в том числе перед программами от довольно крупных компаний ,в результате тестирования было доказано , что программы имеет практическую пользу для предприятий оборонно-промышленного комплекса



СПАСИБО ЗА ВНИМАНИЕ