**Детектирование пожаров.**

## ****1. Мотивация.****

## **Пожар – это беспощадная сила огня, гибель людей, городов, зданий, материальных ценностей, растительного и животного мира. Каждый пожар это единственное, уникальное событие с присущими только ему характеристиками. По количеству и совокупному ущербу пожары прочно занимают лидирующее положение среди всех техногенных чрезвычайных ситуаций (ЧС). Пожары сопровождают человека всегда и повсюду: на земле, под водой, в воздухе, в космосе.**

## **Серьезную опасность для природной среды, экономики и населения представляют массовые природные пожары, которые возникают в лесных массивах, на торфяниках и на степных просторах.**

<https://vniigochs.ru/upload/iblock/1e7/0tiymezc1j8tahbdagbgvbtnkbyg87bf/%D0%94%D0%BE%D0%BA%D1%83%D0%BC%D0%B5%D0%BD%D1%82%204.pdf>

**(«Пожары и взрывы» ФГБУ ВНИИ ГОЧС (ФЦ).**

## 2. Решение.

## Для решения задачи детектирования пожаров планирую использовать библиотеку “TensorFlow спецификации Keras[”](https://developers.google.com/mediapipe/solutions/vision/pose_landmarker), которая позволяет определять/детектировать в кадре интересующие нас объекты/события.

Нейронную сеть планирую обучать по открытым наборам данных от “Roboflow”, “Kaggle” и видео с “Youtube”. Для очистки/фильтрации данных написал свои модули на “Python”, также после автоматической обработки произвел визуальный контроль полученных результатов.

## 2.1.Выбор размера изображений.

Большинство камер наружного наблюдения, которые были предоставлены компанией  
“Квант-Телеком” для стажировки “Безопасный город” имеют разрешение 1920x1080,  
то есть соотношение 16:9 = 1.7778.

Планирую использовать предварительно обученные нейронные сети, такие как “VGG”, “ImageNet” или “ ResNet”, которые были обучены на изображениях с размерами 224x224 пикселей.

В рабочую нейронную сеть будем подавать кадры без преобразования, то есть соотношение сторон, у которых 1.7778. Поэтому перед обучением вырезаем центральную часть кадра по меньшей стороне согласно пропорции 1.7778, а затем сжимаем до размеров 224x224 пикселей, чтобы объекты на которых обучится нейронная сеть и объекты на которых она будет работать имели пропорциональные геометрические искажения.  
  
Оригинальное изображение с камеры 1280x720 пикселей и подготовленное/искаженное изображение для обучения нейронной сети 224x224 пикселей:



Оригинальное изображение с камеры 860x460пикселей и обрезанное по краям до размеров 817х460 (817/460=1.776):



И после этого производим сжатие до размеров 224х224 пикселей:



Трансформирование изображений произвел с помощью модуля “smart\_city\preparation\_utils\crop\_resizer.py”, в котором также реализовал перемешивание по случайному закону перед созданием уникальных имен файлов.

## 2.2. Определение детектируемых классов.

Для классификации и детектирования определяю два класса:  
- 0: “fire” - “Пожар”,  
- 1: “non-fire” - “не Пожар”.

Использовал наборы данных с “Roboflow”, “Kaggle”с двумя классами изображений, а также “Youtube”. Для расщепления видео на кадры использовал модуль“smart\_city\preparation\_utils\frame\_extractor.py”.

## 2.3. Фильтрация размеченных объектов.

С помощью модуля “smart\_city\preparation\_utils\image\_comparator.py”, оставил только непохожие друг на друга изображения, для того чтобы нейронная сеть “не заучивала” данные.

После ручного просмотра/контроля скопировал заданное количество изображений (“fire”, “non-fire”должно быть одинаковым, чтобы набор не был разбалансированным), перемешав и разделив на три папки “train”, “valid”, “test”, также к именам файлов добавил префикс - “f\_”, “nf\_” (“fire”, “non-fire”), выполнил это с помощью модуля “smart\_city\preparation\_utils\image\_tvt\_separator.py”.

В итоге, получил два набора "fire", "non-fire" по 1182 изображений, размер которых 224x224 пикселя.

Ссылка на датасет:  
https://cloud.mail.ru/public/nRgC/X9neA1k4M