**Отчет по анализу данных.**

Дата: Декабрь 2022

Автор: Антонов Александр

**Содержание**

* Утилита для детектирования аномалий adtk
* Метод кластеризации объектов watershed
* Пайплайн работы на проекте
* Методы предобработки изображения: кроп
* Методы предобработки изображения: NlMeansDenoising
* Использование соседних кадров
* Учет статических объектов
* Выбор наблюдаемых значений для построения временных рядов
* Визуализация анализа данных
* Детектирование аномалий на примере детектора человека
* Планы на Январь 2023

**Утилита для детектирования аномалий adtk**

На данном этапе работы при анализе данных решено сосредоточить внимание на построении временных рядов и детектированию аномалий в них.

На практике видится подходящим использование инструмента adkt. Ссылка: <https://adtk.readthedocs.io/en/stable/> Здесь и далее подразумевается использование библиотек на python, если не указано иное. Инструмент использует подходящую для проекта лицензию.

Примеры графиков, построенных при помощи adtk

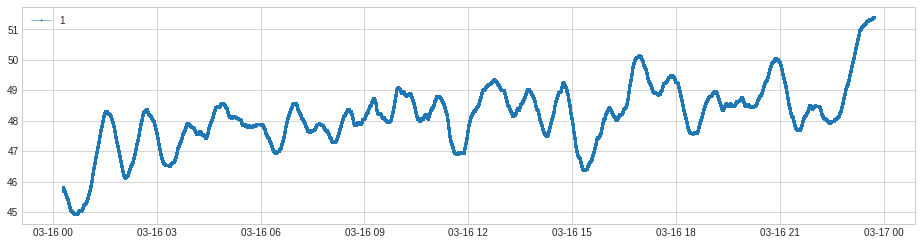


Рис.1

Chart, line chart

Description automatically generated

Рис.2

На рисунке 1 показан синим временной ряд. На рисунке 2 оранжевым выделены зоны аномального поведения. Предполагается, что в дальнейшем на временные интервалы, отмеченные как аномальные, потребуется обратить внимание оператора.

**Метод кластеризации объектов watershed**

Базовые исследования в области анализа изображений при помощи компьютерного зрения решено проводить как решение задачи кластеризации объектов. В частности, используется метод watershed и библиотека opencv. Cсылка: <https://docs.opencv.org/4.x/d3/db4/tutorial_py_watershed.html> Метод watershed использовался на предыдущих стадиях проекта и показал себя эффективным базовым методом. Библиотека opencv подходит с точки зрения лицензии. Имеет реализации на python, что удобно на этапе исследований, и на с++, что будет важно при оптимизации скорости расчетов при использовании в режиме реального времени.

Пример кластеризации

Graphical user interface, application

Description automatically generated

Рис.3

На рисунке 3 слева показано обрабатываемое изображение, куры в виварии. Справа объекты кластеризованы, выделенные курицы отмечены различными цветами.

**Пайплайн работы на проекте**

На одной из декабрьских еженедельных встреч обсуждали, как должен выглядеть пайплайн работы на проекте. Подробнее в презентации: <https://github.com/AlAntonov/poultry/blob/main/report/Alex_20221219_Overview.pdf>

Вкратце пайплайн может выглядеть так

* Детектирование объектов
* Расчет необходимых показателей
* Поиск аномалий на временных рядах

Возможные показатели, которые будем расчитывать

* Количество кластеров
* Размеры кластеров, количество кур в группе, кучность
* Направление кур
* Направления и скорость движения объектов, активность

Аномалии по длительности

* Краткосрочные: Зашел человек
* Долгосрочные: Несколько часов пониженное движение, сон. Экспериментальные изменения, вода, свет, температура.

Аномалии по предсказанности

* Известные заранее
* Обнаруженные алгоритмами

**Методы предобработки изображения: кроп**

В процессе экспериментов заметили, что в некоторых случаях эффективно работать не с целым изображением, кадром с видео, а с его частью, так называемом кропом.

Плюсы:

* Можно работать с областью изображения без статических объектов
* Довольно легко проверить результаты работы алгоритмов, например, подсчитать количество обнаруженных кур
* Сокращается время обработки изображения
* Некоторые рассчитываемые показатели имеют смысл только в части изображения. Например, количество объектов (кур) всегда одинаково, а на части рисунка может меняться во времени, и таким образом имеет практическую пользу для анализа.

A picture containing text, fabric

Description automatically generated

Рис.4

На рисунке 4 левое верхнее изображение – целый кадр, правое верхнее изображение – кроп, часть кадра.

**Методы предобработки изображения: NlMeansDenoising**

Замечено, что на имеющихся видеокадрах изображение кур может сливаться с подложкой, таким образом, затруднена работа алгоритмов детекции объектов.

Как вариант упрощения ситуации – использование алгоритмов подавления шумов. В частности, был испытан алгоритм NlMeansDenoising, в библиотеке opencv использовался метод fastNlMeansDenoisingColored, ускоренная версия алгоритма для работы с цветными изображениями.

На данном этапе можно сделать вывод, что алгоритм в некоторых случаях способствуем лучшему детектированию объектом. Однако часто пользы не приносит. Поэтому решено продолжить поиски хороших алгоритмов. А именно, хотим попробовать Лапласовы пирамиды.