universität innsbruck

Object Recognition

PS - Distributed Systems

David Freina, Mathias Thöni

Project Report

Team members:

- Mathias Thöni
- David Freina

Introduction

- Project number 1
- Programming Language
  - Node.js (mainly)
  - Python (for functions that use OpenCV and scikit)
- Problem motivation
  - ”Smart”-Home security
  - Pet behaviour analysis
  - Wildlife cameras

Workflow structure

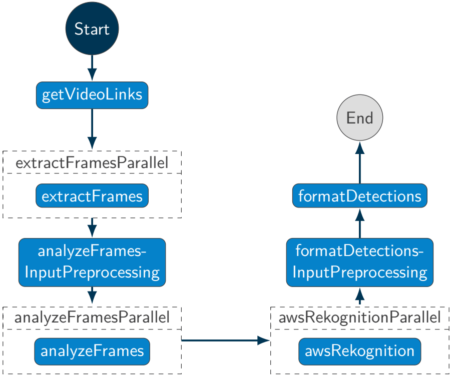

Figure: The FC inspired by FCEditor

Workflow structure

- Data inputs:
  - `videoBucketId` - The id of the S3 bucket with the input videos
  - `numberOfFramesToAnalyzePerInstance` - The number of frames one instance should handle
- Data output:
  - `formattedDetections` - A string containing the information about all detections

Workflow structure

- getVideoLinks - Gets the links to all videos of the given S3 bucket
- extractFrames - Retrieves frames of a video and stores them in split folders on the S3 bucket
- analyzeFramesInputPreprocessing - Prepares the output of extractFrames for the analyzeFrames function
- analyzeFrames - Compares images of a video split folder and deletes same images
- awsRekognition - Calls the AWS Rekognition every video of a video split folder and returns the detections
- formatDetectionsInputPreprocessing - Prepares the output of awsRekognition for the formatDetections function
- formatDetections - Formats all detections to a single human readable string

Main challenges

- Main problems
  - Duplicated detections due to parallelization
- Solutions
  - Still unresolved

Main challenges

- Main problems
  - Duplicated detections due to parallelization
  - AWS Rekognition API Limits
- Solutions
  - Still unresolved
  - FC constraint ”concurrency”

Main challenges

- Main problems
  - Duplicated detections due to parallelization
  - AWS Rekognition API Limits
  - Function dependencies (OpenCV, scikit)
- Solutions
  - Still unresolved
  - FC constraint ”concurrency”
  - AWS ECR and docker images

Main challenges

- Main problems
  - Duplicated detections due to parallelization
  - AWS Rekognition API Limits
  - Function dependencies (OpenCV, scikit)
  - S3 Bucket access restrictions
- Solutions
  - Still unresolved
  - FC constraint ”concurrency”
  - AWS ECR and docker images
  - Two bucket id parameters in FC input for development purposes, removed for submission

Other challenges

- Other relevant challenges
  - Preprocessing inputs for functions
  - Passing through variables in FC
  - Documentation xAFCL

Demo

- Present that your FC works:
  - It works ;)
  - Demo video

Evaluation

Frames per instance	Execution time (ms)
5	127,150
10	102,288
50	86,985
100	84,404
150	106,326

- Sequential will most likely worsen the execution time
- Best performance between 50 and 150 frames per instance

Thank you for your attention!

David Freina, Mathias Thöni