Machine Learning Engineer Nanodegree
Capstone Proposal

Juan José Madrigal Martínez
January 10, 2017

Madrid, Spain
(+0034) 600 86 32 48
juanjomadrigal326@gmail.com
LinkedIn GitHub Kaggle Udacity

Domain Background

This project aims at building a video analysis system for surveillance purposes. It basically finds and indexes the movement-events filmed by a fixed camera.

Video surveillance has become a major and widely used tool for multiple issues [1] and is supported by many companies [2] [3]. But the huge ammount of information which is usually dealt with has led to the need to use Machine Learning techniques to extract patterns, predictions and other refined information. This approach is being implemented [4] and there is much work for Machine Learning engineers to do in this field.

[1] Wikipedia - Surveillance / Cameras

[2] VideoSurveillance.com

[3] Tyco

[4] Briefcam

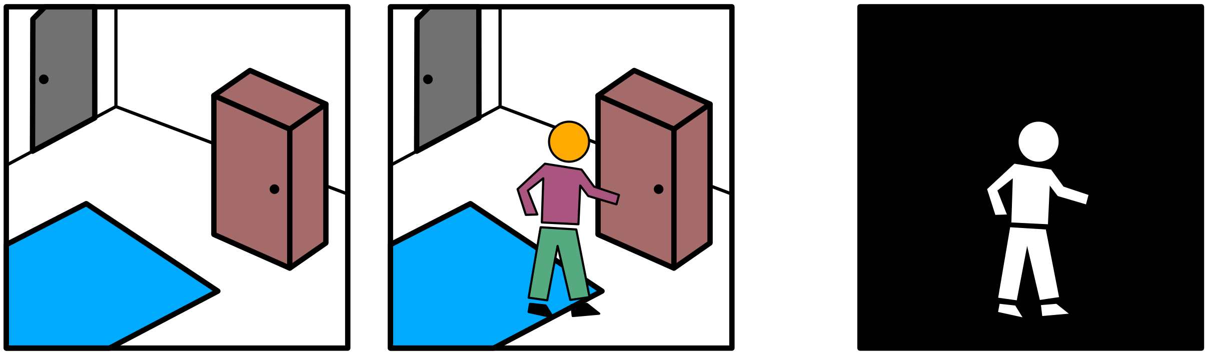
Problem Statement

In video surveillance context, a common problem is having to explore large amounts of videos looking for a particular event, e.g., someone getting in or doing something in particular. Users have to deal with a sequential (and manual) search through the video-source (which can be many hours of stored video) and this is rather inefficient, boring and error prone.

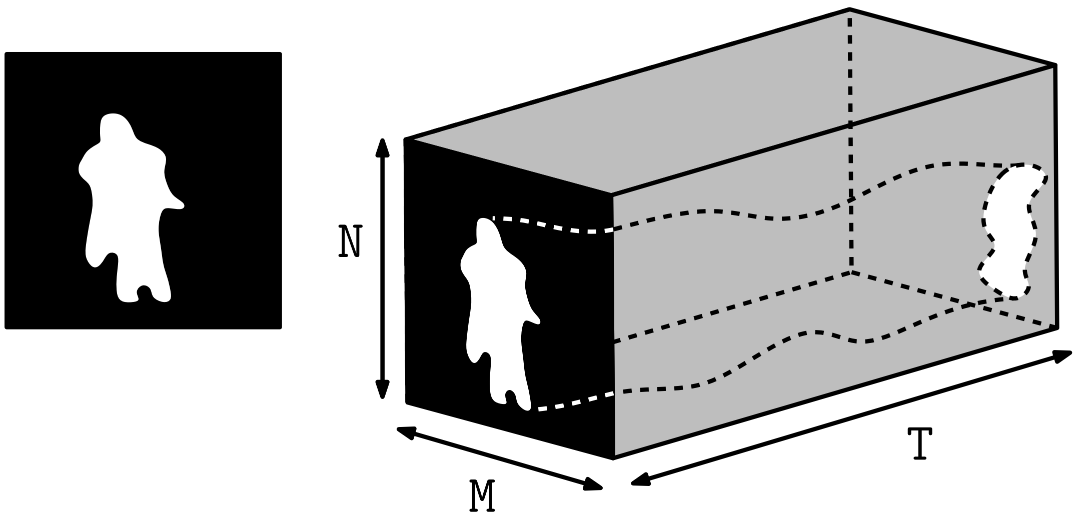
Datasets and Inputs

The datasets and inputs for this project will be obtained after some video preprocessing made on sample videos. The selected videos are located in capstone_project/data/unprocessed_videos and they have been downloaded from VIRAT Video Dataset [1]. The video preprocessing is carried out by the Python files XXXXXXXXX and is not part of the Machine Learning Project.

The video preprocessing takes as input a `.avi` $\mathtt{M} \times \mathtt{N}$ pixels video and a `.png` $\mathtt{M} \times \mathtt{N}$ pixels picture (*background*). For each frame in the video, the frame and the background are compared pixel-wise, and the difference is recorded in a $\mathtt{M} \times \mathtt{N}$ binary array, where 1 denotes difference (above some threshold) and 0 denotes no difference.



If the video has T frames, then the video preprocessing gives as output a $M \times N \times T$ binary array, which is supposed to capture the movement in our scene and that will be the primary input for the project. These arrays are located in XXXXXXXXX



[1] **A Large-scale Benchmark Dataset for Event Recognition in Surveillance Video**, Sangmin Oh, Anthony Hoogs, Amitha Perera, Naresh Cuntoor, Chia-Chih Chen, Jong Taek Lee, Saurajit Mukherjee, J.K. Aggarwal, Hyungtae Lee, Larry Davis, Eran Swears, Xiaoyang Wang, Qiang Ji, Kishore Reddy, Mubarak Shah, Carl Vondrick, Hamed Pirsiavash, Deva Ramanan, Jenny Yuen, Antonio Torralba, Bi Song, Anesco Fong, Amit Roy-Chowdhury, and Mita Desai, *Proceedings of IEEE Comptuer Vision and Pattern Recognition (CVPR), 2011*.

Solution Statement

We propose the use of unsupervised clustering techniques to find the movement-events in the movement-mask space (binary representation: 1 for pixel-moved, 0 for pixel-no-moved). The latter is computed with a foreground subtraction step resulting in a $M \times N \times T$ binary array, for a $M \times N$ video source with T frames.
Every movement event is represented in this 3D space as a 'volume' or 'body' (contiguous '1' cells of movements), and the unsupervised clustering algorithm is supposed to successfully segment most of them.
A web application is envisaged to be built, receiving video files and delivering the list of detected events linked to the corresponding point in the original video source.

Benchmark Model

In this section, provide the details for a benchmark model or result that relates to the domain, problem statement, and intended solution. Ideally, the benchmark model or result contextualizes existing methods or known information in the domain and problem given, which could then be objectively compared to the solution. Describe how the benchmark model or result is measurable (can be measured by some metric and clearly observed) with thorough detail.

Evaluation Metrics

In this section, propose at least one evaluation metric that can be used to quantify the performance of both the benchmark model and the solution model. The evaluation metric(s) you propose should be appropriate given the context of the data, the problem statement, and the intended solution. Describe how the evaluation metric(s) are derived and provide an example of their mathematical representations (if applicable). Complex evaluation metrics should be clearly defined and quantifiable (can be expressed in mathematical or logical terms).

Project Design

In this final section, summarize a theoretical workflow for approaching a solution given the problem. Provide thorough discussion for what strategies you may consider employing, what analysis of the data might be required before being used, or which algorithms will be considered for your implementation. The workflow and discussion that you provide should align with the qualities of the previous sections. Additionally, you are encouraged to include small visualizations, pseudocode, or diagrams to aid in describing the project design, but it is not required. The discussion should clearly outline your intended workflow of the capstone project.