Crowd Counting
Assignment 10 for CS 35L, Winter 2017

Nikola Samardzic
nikola.s@ucla.edu
UCLA Undergraduate Student

Abstract

This article is a review of recent developments in Computer Science and Artificial Intelligence that contribute to better quality Crowd Counting (CC) algorithms. The Crowd Counting Problem is concerned with automatic estimation of crowd sizes at particular events.

1 Approaches

There are multiple methods of Crowd Counting. Firstly, a distinction must be made between algorithms that rely on images and those that rely on video content. Also, some approaches require image data from multiple angles, spaced between many hours to precisely estimate crowd size, while others perform head-counting by using only one image. Another big issue for CC is privacy of the people being counted. Hence, some papers attempt to approach the problem by focusing on motion detections rather than head counting algorithms which compromise privacy.

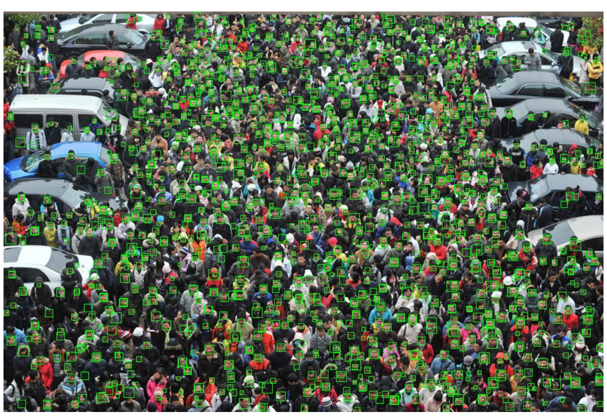

Figure 1: Head-detection software can be used for CC. It is very versatile as it works well even in dense crowds. [3]

One of the main problems in solving these problems is creating a general solution - one that would work for pedestrians moving on the street, small group gatherings, sporting events, concerts, parties, and many other events and locations many of which induce very different behaviors of the participating crowds.

Also, since information about the crowd is fed in through either video or picture, many geometrical problems arise. These include perspective, camera focus, and proper detection of venue borders.

2 Overview

All CC algorithms are based on similar grounds. In the core, they all rely on either Convolutional/Regular Neural Networks or Simple Linear Regression methods. The real differences in these algorithms are defined by the parameters that are fed into the learning mechanism. Now we will outline a method of CC presented by the researchers at The Queensland University of Technology in Brisbane.

2.1 CC using Multiple Local Features

This specific approach relies on input from one fixed camera video recording, although results may be improved if multiple cameras are used.

The main idea behind the algorithm consists of following steps:

1. Extract foreground of image
2. Separate foreground into groups (called blobs) by proximity and motion patterns
3. Estimate number of people in each blob
4. Return the sum of these estimates over all blobs

The algorithm extracts the foreground by noticing which pixels change and which don't. Tracking the moving pixels gives a probability distribution on whether a specific pixel is in the foreground. After this step, the foreground is separated into multiple groups of connected foreground pixels called blobs.

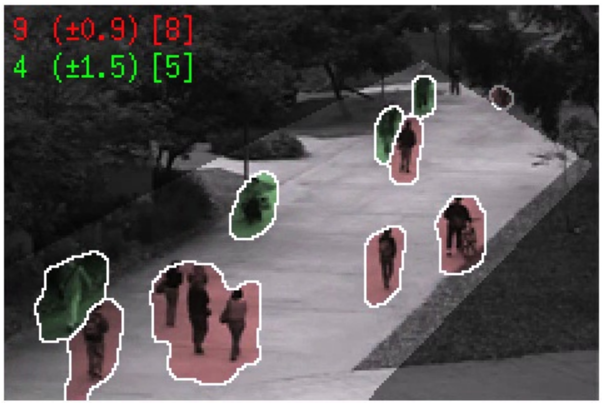

Figure 2: Illustration of the process of extracting foreground blobs from an image. [1]

Using either a Neural Network or a Linear Regression Model the number of people within each blob is estimated. The parameters fed into the model include: blob area, blob perimeter, blob complexity (ratio of perimeter to area), number of edges on perimeter, and angles of perimeter edges. After this, we just return the sum of the count estimates for each of the blobs to arrive at an estimate of the total crowd size.

Figure 3: Red and blue blobs contain about the same amount of people, yet the red blob is significantly smaller than the blue one due to perspective. [4]

Another notable problem is that of perspective. As one would expect, best results are obtained when using recordings from a birds'-eye view camera. However, most of the time, the camera's used for CC are installed 2-5 meters above the ground. The recordings from these cameras might confuse the algorithm due to perspective. People in the distance are going to appear to be smaller than people nearby and this has to be accounted for by the algorithm. Fortunately, it is not very hard using linear algebra to figure out from which angle the camera is recording and add appropriate weights to pixels on the screen that solve the problem.

2.2 Results of Model

The performance of this approach is presented in the following table:

Model	Error	MSE	Accept.
Proposed	1.353	3.065	95.67%
Holistic	1.662	4.028	94.00%

Proposed: The proposed algorithm that relies on separating the foreground into blobs.
Holistic: Previously used algorithm that utilizes similar features but applies them to the whole image.
Error: Mean value of the difference between the estimate and ground truth
MSE: Mean value of error squared
Acceptability: Percentage of frames for which the algorithm's absolute error is less than 3.

3 Possible Applications

I discovered this topic on ACM News. It was probably popular because of the controversial statements of the Trump Administration that Trump's inauguration was the most visited in history. Clearly, quality CC methods could have resolved the dispute. More generally, it is worth considering how programming can be further applied to aid journalism in the pursuit of presenting truth to the general public.

Also, the Machine Learning (ML) methods used in CC algorithms are also widely applied to other fields. In fact, CC method only further validates the fact that ML can be used to train solutions to very complex and multifaceted problems that would be hard to implement explicitly. As exemplified by past breakthroughs in computer science, algorithmically solving problems that seem currently to require what laymen would consider a 'human touch' will probably depend highly on further developments in ML. This is because ML provides computers with an ability of processing information in a very special way – using Neural Networks that mimic the behavior of human neurons.

More specifically, advances in CC software might provide other services to the public. Since CC is related to monitoring human motion, it can be used to notice thieves in stores automatically. Similarly, it can be applied to providing security in densely populated areas, noticing suspicious behaviors that require immediate action by the police or calling ambulances for people experiencing health emergencies on the street. The algorithms can also be modified to aid traffic law enforcement by spotting red-light violations, improper lane-switching, and others. Notably, CC would be instrumental in providing safety for large crowds, especially in concerts or protests, which can easily get violent and dangerous. Also, it could help monitor traffic congestion in real time. Since these programs can keep track of motions of crowds, they can also be geared towards spotting specific people that wanted by the police, for example. This raises many privacy issues that will definitely surround CC in the future.

As always, any advances in AI are met with scepticism in people who fear it would enforce a 1984-like police-state. Indeed, their fear is valid, especially for CC, as it could possibly provide the government with a lot of information that most people would consider strictly private. Therefore, keeping privacy issues in the back of our heads is crucial if we plan to create AI tools. CC and others are powerful algorithms, but we unfortunately have no guarantee that they will not fall into the wrong hands.

References

[1] David Ryan, Simon Denman, Clinton Fookes, Sridha Sridharan. Crowd Counting using Multiple Local Features. Digital Image Computing: Techniques and Applications, 2009

[2] Antoni B. Chan, Zhang-Sheng John Liang, Nuno Vasconcelos. Counting People without People Models or Tracking. University of California, San Diego

[3] Venkatesh Bala Subburaman, Adrien Descamps, Cyril Carincotte Counting people using generic head detector. IEEE Ninth International Conference on Advanced Video and Signal-Based Surveillance, 2012

[4] Pietro Perugino The Delivering of the Keys. Sistine Chapel, Rome, 1481