Mobile Crowd Mapping

Jukka M. Krisp[1], Alan Kwok Lun Cheung[2]

[1] Cartography, Technical University Munich (TUM), Arcisstr. 21, 80333 Munich, Germany; [2] School of Geography and Environmental Science, University of Auckland, 10 Symonds Street, Auckland, New Zealand

Abstract. Within this paper we investigate the possibility to have a tool that counts individuals within a static crowd photo, which is taken with a camera included in a smart phone. These individuals form different density patterns. We explore ways to map and display the density information on top of the photo on the smart phone screen. This enables the user to have a fast estimate about how many individuals are within the crowd and how is the underlying density distribution within the crowd. This paper includes a case study within a student group photo in an outside environment.

Keywords: Location Based Services (LBS), crowd mapping, visualization, mobile phone app

1. Introduction & Background

Crowds are highly dynamic where density distribution varies even within a crowd.. It is especially critical for security and rescue services to monitor crowds' behavior. Additionally there are less critical situations in which crowd mapping, for example can be used in group picture situations or the usage in classrooms or lecture halls. Within this paper we focus on a student group picture situation which is similar to a classroom scenario. Therefore as a note, the framework for this work-in-progress paper documents a somewhat limited study. Within a broader context, computational crowd mapping tools have a wide range of applications. In crowd management, it is difficult for the police or fire and rescue services to spot areas of high density within a crowd. A counter tool for individuals within a crowd and an underlying computational functionality to estimate the density distribution within a crowd may be a very valuable tool. Application of this functionality on a smart phone or a tablet allows it to be universally available and enables the adaptation of this tool to a wide user base.

As Junior (2010) points out, "it is interesting to note that the problems of people counting and tracking are related, since both of them have the goal of identifying the participants of a crowd. However, the counting problem usually requires only an estimate of the number of people, regardless of their position (and temporal evolution)" (Junior, et al., 2010). Within this work we deal with static images taken with a mobile phone camera, therefore we are focusing in the static part of counting the number of individuals within a crowd. The monitoring and analysis of crowds have been investigated by a number of researchers. As stated by Krisp et al (2012), "Big events in various situations like festivals, concerts, sport events or religious activities receive an increasing number of attending people, which makes crowd monitoring indispensable especially for security reasons. An overview of several crowd monitoring and analysis techniques is given in Zhan et al. (Zhan, Monekesso, Remagnino, Velastin, & Xu, 2008), including tracking and observation methodologies as well as approaches for crowd modeling and event inference. Depending on the number of monitored people and the data acquisition device, either the crowd itself or single people within the crowd can be identified." (Burkert, Schmidt, Butenuth, & Hinz, 2010; Krisp, et al., 2012).

1.1. Problem statement

In many classrooms the number of students exceeds a threshold which could be counted with a single. With an increasing number of students filling the classroom, it is increasingly difficult to "see" and estimate the number of individuals sitting in front of the teacher or lecturer. In a typical situation, students usually cluster in a room, for example with some sitting in the front, some cluster at the back and perhaps another cluster close to the door. As another example

Figure 1 shows a group photo situation in an outdoors environment. In this case the photo shows a group of students, which seem rather equally distributed among space. Also in this case it is difficult to estimate the number of individuals within this crowd and it is challenging to "see" the underlying density pattern formed by this crowd.

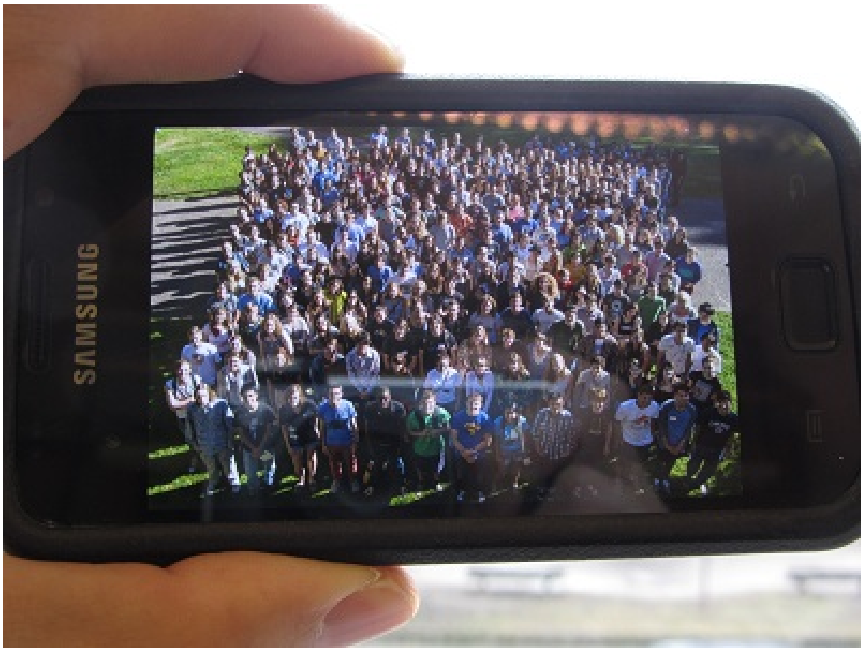

Figure 1: A picture taken with a smart phone camera showing an atypical crowd with students rather equally distributed among the environment

Krisp et al point out, that "crowds are scattered, meaning that partly the individuals are close together and partly there is more open space between the individuals. These form a pattern with different densities within the crowd. How can we display and visually recognize the static and dynamic densities (and the density changes) within the movement patterns?" (Krisp, et al., 2012). Density visualization, for example displayed on an interactive interface on a mobile display helps the identification of the number of individuals within this crowd and the visual investigation of crowd patterns.

1.2. Aim / Focus

Our test case is related to a very common situation within the university teaching situations. The teacher or lecturer entering a lecture room would like to know, how many students are attending this course? To count the students is possible, but usually takes too much time and estimating the number of students will give just a rough number.

Here we investigate the computation and communication of crowd density information from a group picture taken in an outside environment. The same is currently tested within a lecture hall. The teacher or lecturer takes a picture of the crowd with his / her smart phone and the number of individuals is mapped and displayed on the screen. Further analysis considers the individuals sitting in a lecture hall (or classroom) that form specific distribution patterns and different densities of individuals within space. How can we provide the user, in our case a teacher or lecturer in a classroom, an easy tool to count the number of student sitting or standing in front of the user?

2. The crowd counter app-tool

As Junior (2010) summarizes, "there are several models developed to estimate the number of people in crowded scenarios using computer vision techniques, divided into three categories, (1) pixel-based analysis, (2) texture-based analysis, and (3) object-level analysis" (Junior, et al., 2010). Within this work we focus on the object-bases analysis, in particular in the individual faces within in a crowd.

We investigate a tool to recognize these distributions and to provide an interface that can be used in a mobile phone (Android app) to detect and visualize the densities of crowds. Visual mobile displays and interactive techniques are combined with computational processing, which enable the analysis of larger crowds. That would not be possible with purely visual methods and without the help of computational methods.

As a basis for this "mobile crowd mapper", we produced an app-tool that helps us to count the number of students when they are gathered as a group either in an outside environment or in a classroom situation. We use mobile devices with cameras to capture an image of the ambient environment, including the crowd. A standard face-detection algorithm is then run on the image, producing a count of people in this environment. Finally, we calculate the density and distribution of the crowd in this particular environment. The result is visualized on the screen of the mobile device as overlaying density maps.

The crowd counter app relies on mobile devices such as smartphones that satisfies the following requirements: 1) Capture a photo of reasonable size (5 Megapixels+), 2) Has enough computing power to run face-detection algorithms, output rendering and geometric corrections operation and 3) Visualize the output.

For our prototype, the Android platform is chosen as a basis for the crowd-counter app. The flexibility of Android API and the amount of community support means that apps can be created within reasonable amount of time and difficulty.

The testing version of the app was built on internal face-tracking API available on Android version 4 or above. The detection, calculation and visualization were to be done on-the-fly on screen. However since the built-in algorithm of Android was designed for camera apps rather than large-scale face detection, its detection efficiency was low. As a result, real-time tracking of large crowd was not possible.

In order to optimize the routine rather than focusing on real-time visualization, we adopted a semi real-time routine where a photo is taken and stored onboard the smartphone before analysis. At the backend, OpenCV for Android (OpenCV, 2013) is used in place of the internal algorithm. OpenCV stands for Opensource Computer Vision, which is a popular computer vision library including face detection algorithms. The detection is of objective-level analysis as suggested by Junior et. al. (2010). This version of OpenCV utilized in this research is tailored for low power Android devices.

After the detection phase, a density map is calculated from the distribution of the detected individuals. The result is subsequently rendered on top of the original photo using Android API's internal functions. There are environmental settings that require geometric correction in order to accurately calculate the distribution of individuals. For example, provided that people sits in regular spacing, individuals sitting towards the back of a lecture hall would appear closer together on an image than individuals at the front, producing a paradox that density at the back is higher than the front. Realted correction techniques had been widely applied in Remote Sensing applications(Jensen, 1996). Incorporation of these techniques will be in our future research.

3. The app test in an outside group environment

An application test within an outside environment picture documents the clustering of students. The density information can be processed further to investigate possible temporal changes or in an inside environment the plan usage of lecture halls and classrooms. The results acquired can possibly be used to extend and enhance the algorithms and mapping functionalities to be used in outdoor settings and in "more chaotic" environments than lecture halls or classrooms, for example for crowd management in safety and security planning. The detection algorithm still includes a number of mismatches, meaning the individuals that are not detected (in this sample case 77 individuals were not detected).

Figure 2 shows the application of the standard detection algorithm applied to our test image. In this case there are 322 people in total in the image. From those 248 detections were made, within those 245 are correct detections, 2 are misdetections and 1 is a double detection. 77 people were not detected. Additionally it is interesting to notice that the non-detections are aggregated at the back. Lighting condition and shadows do affect the detection.

The number of detected individuals can be improved by enhancing and extending the parameters that are used in the face-detection algorithm. Still the number of mismatches depends on the picture at hand and on the environment the picture is taken in. With further advancements in face detection algorithms, it is safe to say that the number of mismatches will decrease.

Figure 2: A mock-up illustration of the detected individuals in a crowd within an outside group picture, including mismatches

Visualizing densities in the crowds

The densities are based on the proximities of the different recognized faces. To effectively show the density we impose a density overlay onto the photo taken with a phone camera, taking into account the ever changing viewing perspective. This challenge of a changing field of depth and angle can be solved by using the geometry of the clusters of detected faces, in our case the size and boundary of all the detected faces of individuals in the sample crowd. Junior states that "an important problem in crowd analysis is people counting/density estimation (either in still images or video sequences). For instance, crowd density analysis could be used to measure the comfort level in public spaces, or to detect potentially dangerous situations" (Fruin, 1971; Junior, et al., 2010).

The density calculation is based on the kernel density function (KDE) with a bandwidth based on the average distances between individuals sitting in the lecture room. This bandwidth can be interactively changed to study the distribution of the crowd from the location of individuals to a generalized map that provides an overview for high and low densities. The temporal domain can be realized also by taking multiple photos of the same setting. The resulting density maps efficiently communicate varying distribution patterns.

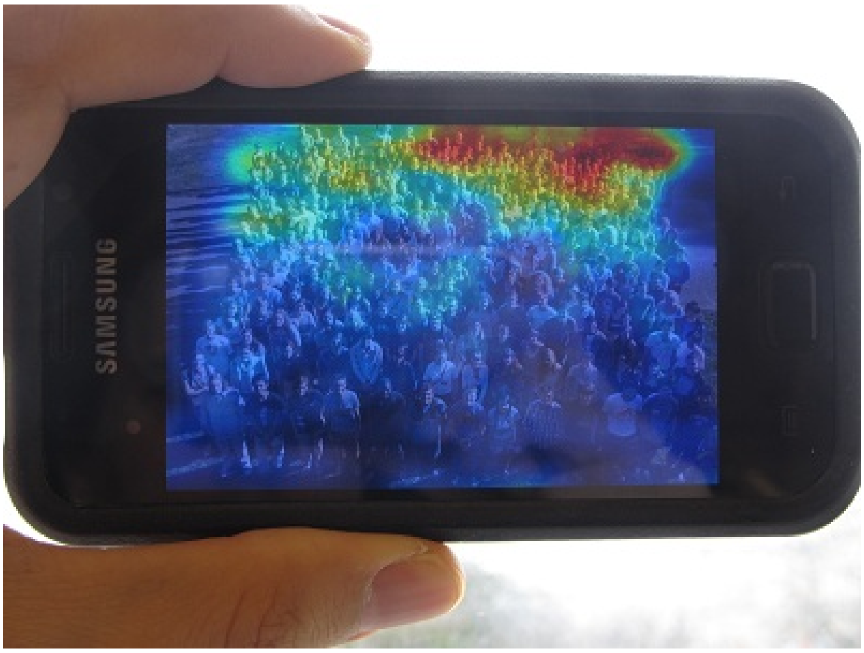

Figure 3: A mock-up depiction of a possible density map of detected individuals is overlaid on top of the original image, based on the detected individuals and the need to have a rough account of the in depth information in the picture

Figure 3 shows a mock-up illustration of an intended result. The individuals are detected and the density information is displayed on top of the crowd or in this case the individuals. The in depth information in the picture needs to be taken into account. In this case it appear that the individuals are standing closer together in the upper right corner of the picture, which is indeed the case. The overlay density "map" makes it easier to detect hotspots within the crowd of individuals and to "see" the distribution patterns of individuals within the environment.

4. Conclusion & Challenges

The detection results are reasonably accurate, in terms of misdetections. If the detection algorithm is relaxed, we will detect more people but also increase the misdetections.

The geometry of images, for example the coverage and depth of view, needs to be solved with geometric correction. Apart from that, efficient face detection algorithms which could detect side-facing people or people with accessories will enhance the analysis. Finally and perhaps most difficult is to rectify and compensate geometry of a crowd where people locates at terrain of different elevations.

Within this study we have documented the idea of having a counter tool for individuals. This idea needs to be tested within a lecture or classroom environment. The implementations is based on standard face recognition algorithms and implemented into an off-the-shelf Android smart phone. Further investigations are directed towards the density recognition and visualization within the picture of the crowd that is taken with a smart phone. Especially the cartographic design of the resulting density picture needs further investigations.

References

Burkert, F., Schmidt, F., Butenuth, M., & Hinz, S. (2010). People Tracking and Trajectory Interpretation in Aerial Image Sequences. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 39* (3).

Fruin, J. (1971). *Pedestrian and Planning Design*. Metropolitan Association of Urban Designers and Environmental Planners, Elevator World Inc.

Jensen, J. R. (1996). Introductory digital image processing: a remote sensing perspective.

Junior, J., Musse, S., & Jung, C. (2010). Crowd Analysis Using Computer Vision Techniques. *Signal Processing Magazine, IEEE, 27* (5), 66-77.

Krisp, J. M., Peters, S., & Burkert, F. (2012). Visualizing Crowd Movement Patterns Using a Directed Kernel Density Estimation. In M. L. Krisp J.M., Pail R., Stilla U. (Eds.) (Ed.), *Earth Observation of Global Changes* (pp. 255-269). Springer, Berlin.

OpenCV (2013). *OpenCV4Android SDK.* Available at http://opencv.org.

Zhan, B., Monekesso, D., Remagnino, P., Velastin, S. A., & Xu, L. (2008). Crowd Analysis: A Survey. *Machine Vision and Applications, 19* (5-6), 345-357.