Crowd Detection from Still Images

Ognjen Arandjelović

Department of Engineering, University of Cambridge, CB2 1PZ, UK

oa214@cam.ac.uk

Abstract

The analysis of human crowds has widespread uses from law enforcement to urban engineering and traffic management. All of these require a crowd to first be detected, which is the problem addressed in this paper. Given an image, the algorithm we propose segments it into crowd and non-crowd regions. The main idea is to capture two key properties of crowds: (i) on a narrow scale, its basic element should look like a human (only weakly so, due to low resolution, occlusion, clothing variation etc.), while (ii) on a larger scale, a crowd inherently contains repetitive appearance elements. Our method exploits this by building a pyramid of sliding windows and quantifying how "crowd-like" each level of the pyramid is using an underlying statistical model based on quantized SIFT features. The two aforementioned crowd properties are captured by the resulting feature vector of window responses, describing the degree of crowd-like appearance around an image location as the surrounding spatial extent is increased.

1 Introduction

In this paper we are interested in detecting and segmenting out crowds of humans in still images. Given an image, the goal is to determine if there is a crowd in it and if so, which portions of the image it occupies.

Being able to infer the presence of a crowd in an image is a useful task in itself: the formation of crowds can cause delays in underground passages, shopping centres and streets, or be an indication of civil unrest. In the automotive industry, crowds are of interest as a potential road hazard. Additionally, crowd segmentation is a necessary preprocessing step that precedes a higher level task, such as counting (or, more generally, estimating) the number of individuals in crowd, or analyzing their behavioural dynamics and interaction. The areas to which crowd detection can thus be applied to range from psychological research and macro-engineering, through to crime prevention and detection.

Problem formalization and difficulties. We can define a crowd as a group of spatially proximate objects of a certain class. In this work we are specifically considering human crowds, as the type that is usually of most interest in practice.

There are several reasons why crowd detection is challenging. Firstly, limited resolution of images means that the evidence for a single person is usually rather scarce. Given that partial occlusions are abundant in crowds, and the variation in clothing, pose and

BMVC 2008 doi:10.5244/C.22.53

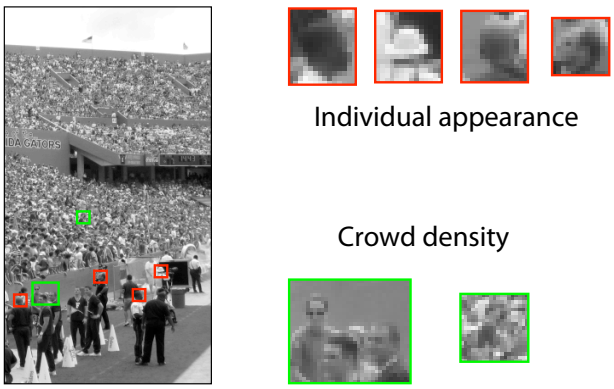


Figure 1: Crowd detection is made difficult by variation in individual appearance, caused by changes in clothing, pose, illumination and scale, as well as by low resolution, partial or full occlusion, and crowd density.

illumination rather extreme (see Fig. 1), detection of individuals as the basic building element is not a promising approach [6]. On the other hand, an approach that directly looks for multiple people, faces problems of modelling a much increased range of variability in their *combined* appearance, as well as crowd specific factors such as the spacing of individuals in the crowd i.e. its density.

1.1 Previous work

In contrast to the related problems of finding humans in images, such as face [8, 10, 11] or pedestrian detection [9, 2], there has been comparatively little work done on crowds. What is more, in most of the previous research the problem of robust crowd detection is almost entirely avoided by using some simple form of background subtraction. Roqueiro and Petrushin [7], for example, use a static camera and data acquired over a long period of time to estimate background appearance. Brostow and Cipolla [1] apply independent motion detection to crowds, effectively performing motion segmentation on a sparse set of interest points. This approach is clearly limited by the lack of an appearance model and thus not able to find static individuals (giving rise to false negative errors) nor recognize when moving objects are not humans (giving rise to false positive errors). Furthermore, the use of independent motion for counting humans in crowds is questionable, as crowds (or parts thereof) often exhibit a degree of behavioural coherence. Rabaud and Belongie [5] propose a similar approach which suffers from virtually the same limitations. The main difference is the use of a weak geometric model, a bounding box, which constrains the spatial extent of each independently moving body (i.e. group of interest points) while allowing for articulation within it (also see [3]). This comes at the cost of high scale dependence, as the authors do not suggest a way of automatically choosing the bounding box size. A different set of assumptions is made by Reisman *et al.* [6] in their system specifically designed to detect crowds of pedestrians. They rely on the detection of zebra crossings and left-to-right (or vice versa) motion of pedestrians of interest, relative to a forward-facing camera on a moving car.

In contrast to the aforementioned methods, the algorithm proposed in this paper does

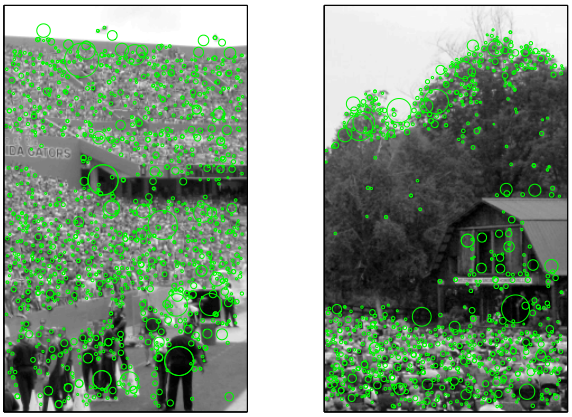


Figure 2: Detected interest points in two images containing human crowds. The radius of a circle used to represent a point shows its characteristic scale. It can be seen that the granulated nature of crowds universally produces a high number of interest points.

not perform background subtraction. We also do not use motion or video, and instead rely purely on appearance cues and single images. Finally, our approach is not based on detecting individual persons and can thus be applied to both small and large crowds.

2 Method details

The concept of a crowd inherently involves repetitive occurrence, which necessitates a certain spatial extent over which this repetition is exhibited. This is why in order to propagate local information, on the highest level we approach segmentation as a minimal graph cut problem. In the sense that the graph vertices correspond to actual image pixels (and their adjacency to that in the image), the proposed method is dense in nature. However, our approach has sparse characteristics in that appearance is described in terms of a sparse set of local features. Extraction and modelling of these is the first step of our algorithm and is addressed next.

Basic features. On the lowest level we use local features to characterize image content. These correspond to a sparse set of interest points, detected as scale-space extrema in a difference of Gaussians pyramid constructed from the original image. By their very nature, crowds universally produce a large number of interest points, as shown in Fig. 2.

Similarly to Lowe in [4], we use the SIFT descriptor to describe each point's neighbourhood, at the scale at which the interest point was detected. We quantize each descriptor by assigning it to the nearest of the K clusters, or SIFT words, estimated by K-means clustering descriptors extracted from a training image set (we used $K = 1000$). Our main contribution concerns the manner in which the obtained set of SIFT words is employed.

2.1 Constant scale model

Let us assume for a moment that we are dealing with crowds at a fixed scale. We wish to decide whether a particular location in an image (note that this is not necessarily the location of an interest point) corresponds to a crowd or non-crowd region. To do this, we consider a rectangular patch around it and seek to quantify how "crowd-like" it is.

Our approach is to assume that the expected number of detections of the i-th SIFT word in an image patch of a specific size is $\lambda_i^{(p)}$, if the patch is a crowd patch i.e. if it corresponds to a region in the image where a crowd is present. Since the number of detected interest points (and this extracted SIFT words) is generally small relative to the number of image pixels, a suitable model for predicting the probability of observing k_i instances of the i-th word is the Poisson distribution. Thus we can write:

$$p\left(k_i|\text{crowd}\right) = \frac{e^{-\lambda_i^{(p)}}\left[\lambda_i^{(p)}\right]^{k_i}}{k_i!}. \tag{1}$$

Furthermore, we assume statistical independence between the counts of any two words in a crowd patch:

$$p\left(k_i,k_j|\text{crowd}\right) = p\left(k_i|\text{crowd}\right)p\left(k_j|\text{crowd}\right). \tag{2}$$

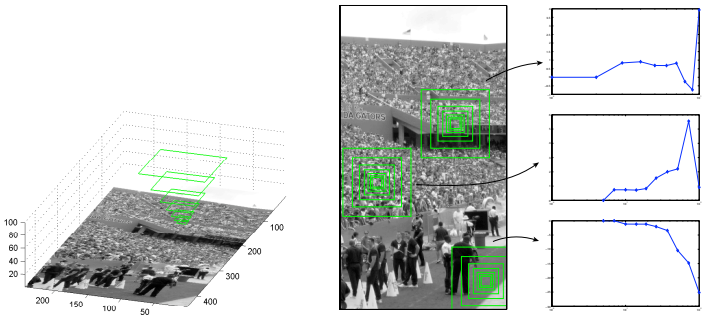
Using the same model for non-crowd patches, but now with the Poisson parameter $\lambda_i^{(n)}$, the log of crowd and non-crowd model likelihood ratios is:

$$\mu = \log p\left(k_1,\ldots,k_K|\text{crowd}\right) - \log p\left(k_1,\ldots,k_K|\text{not crowd}\right) = \sum_{i=1}^{K}\left\{\lambda_i^{(n)} - \lambda_i^{(p)} + k_i(\log\lambda_i^{(p)} - \log\lambda_i^{(n)})\right\}. \tag{3}$$

Sliding window pyramid. Even under the assumption of uniform crowd scale (which we shall abandon in the next section), the application of the proposed statistical model is made difficult by the choice of the spatial extent of the sliding window. Specifically, the problem lies in the inherent tradeoff between spatial accuracy and discriminative information content. Consider the set of patches corresponding to all possible placements of the sliding window. The smaller the window size is, the higher localization accuracy is achieved, since a smaller proportion of patches contains both crowd and non-crowd pixels. On the other hand, as window size is increased, so is the number of SIFT words that fall within it. This means that more evidence is present that can be used to infer the corresponding image content.

To exploit the benefits of different window sizes, instead of examining only a single one, at each pixel we consider a *pyramid* of patches centred on it, see Fig. 3 (a). The likelihood ratio of (3) is then computed for each of them. To do so efficiently, we formulate our model in terms of the average number of detections of the i-th SIFT word per pixel, i.e. its *image density*. By doing this, we effectively exploit the repetitive nature of a crowd. Assuming that the density of the i-th SIFT word in crowd regions is $\rho_i^{(p)}$ and in non-crowd regions $\rho_i^{(n)}$, the log of likelihood ratio of (3) for a square window of size $(w\times w)$ can be written as:

$$\mu(w) = \sum_{i=1}^{K}\left\{\rho_i^{(n)}w^2 - \rho_i^{(p)}w^2 + k_i^{(w)}(\log\rho_i^{(p)} - \log\rho_i^{(n)})\right\}, \tag{4}$$



(a) (b)

Figure 3: (a) A pyramid of image patches of increasing spatial extent is implicitly constructed at each image location. (b) The obtained variation of likelihood ratios of crowd and non-crowd models can be used to infer the local image content: (from top to bottom) boundary of crowd and non-crowd regions, pure crowd and pure non-crowd.

and the result over the entire pyramid for windows of sizes $[w_1,\ldots,w_N]$ becomes the N-tuple $\mathbf{r} = [\mu(w_1),\ldots,\mu(w_N)]$. In our implementation, a set of 10 sizes was used, uniformly spaced on the logarithmic scale between $w_1 = 5$ pixels and $w_{10} = 100$ pixels.

2.2 Scale-invariant word density

In the discussion so far, we considered crowds at a fixed scale. In general, the image scale of individuals in a crowd will vary with scale, and so will the expected density of a particular SIFT word. However, it is not the variation in scale between images that presents the greatest challenge, but rather that of individuals in a crowd *within a single image*. This is the case both when model parameters are inferred during training, as well as when the learnt model is applied on novel data.

We solve this by using the *scale-invariant word density*, $\hat{\rho}$. Let's say that in the training stage, n_i instances of the i-th SIFT word were detected over an image area A. The average density of the word, as used in (3) is simply:

$$\rho = n_i/A. \tag{5}$$

On the other hand, we define the corresponding scale-invariant word density as:

$$\hat{\rho} = \frac{1}{A}\sum_{j=1}^{n_i} 1/s_i^{(j)} \tag{6}$$

where $s_i^{(j)}$ is the characteristic scale of the j-th detection of the i-th SIFT word. We use $\hat{\rho}$ in the place of ρ in (3), with the additional modification that the observed words are also counted in a scale-normalized fashion:

$$\hat{k}_i = \sum_{j=1}^{k} 1/s_i^{(j)}. \tag{7}$$

Likelihood ratio obtained in this manner, that is to say, its variation over space and levels of the sliding window pyramid, are shown for an example image in Fig. 4.

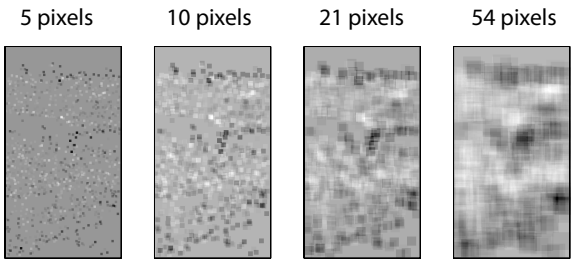


Figure 4: Likelihood ratios of crowd and non-crowd appearance models using sliding windows of varying spatial extent, represented as images. Brighter pixels indicate more crowd-like appearance of the corresponding image patch (centred at the pixel in question). As expected, the reliability of decision is lower when smaller patches are used (the result is noisier). On the other hand, spatial localization is less precise for larger patches (the result appears blurred).

2.3 Characteristic responses

In the previous section we showed how by analyzing a pyramid of image patches of increasing spatial extent we produce a 10-dimensional feature vector, "a crowd response profile", at each pixel. This response profile is rich in information about local crowd and non-crowd content. This is illustrated in Fig. 3 (b) on three examples. In the first, top-most example, the response is noisy and oscillates around zero. This consistent lack of preference for both the crowd or non-crowd models is indicative of a crowd/non-crowd boundary location. This indeed is the case, as inspection of the original image shows. The second example is the typical response at a location in the crowd, which shows increasing decision confidence as the spatial extent is widened. A similar profile, but opposite in the sign, is seen in the last example, of a distinctly non-crowd location.

The type of possible responses is clearly much larger than the three shown. For example, high crowd-like appearance can be observed up to a certain window width, followed by a decline, indicating a small group of people (relative to the maximal window size). What we wish to do is learn how the response profile $\mathbf{r}$ relates to the probability that the corresponding image location belongs to a crowd segment in the image $p(crowd|\mathbf{r})$. We achieve this by training a radial basis function SVM using $20,000$ responses obtained and randomly chosen from a training set of 15 images. Fig. 5 (a) and (b) show the output of a trained SVM using a previously unseen image.

Propagating local information

The SVM output obtained at the last stage of our algorithm is expectedly noisy. This is a consequence of both the extent of appearance variation within a crowd, as well the coarse nature of our features. The final stage of our algorithm uses redundant information from overlapping sliding windows, to obtain a spatially consistent segmentation. We formulate segmentation as an energy minimization problem solved using Graph Cuts:

$$E = \sum_{l} c^{(1)}(l) + \sum_{l_1} \sum_{l_2 \in n(l_1)} c^{(2)}(l_1, l_2) \tag{8}$$

where $c^{(1)}(l)$ is the cost of assigning the positive label to the pixel l and $c^{(2)}(l_1, l_2)$ the cost of assigning the same label to neighbouring pixels l_1 and l_2. The choice of the

(a) (b) (c)

Figure 5: (a) Raw classification result (white corresponds to crowd i.e. positive output, black to non-crowd i.e. negative output) and (b) the associated probability of crowd across image, using a radial basis function SVM. (c) Final segmentation output, using Graph-Cuts.

pixel-wise cost $c^{(1)}$ is straightforward - we use the SVM output of the previous section, $c^{(1)}(l) = 1 - p(crowd|\mathbf{r}(l))$. On the other hand, costs $c^{(2)}$ are assigned using the output of the sliding window pyramid. As before, consider an image patch and the associated log of the likelihood ratio of crowd and non-crowd models. A large magnitude of the log-likelihood is indicative of evidence consistent with a particular model across the entire patch. A log-likelihood of near zero, as already discussed on an example in Fig. 3 (b), means that the patch contains both crowd and non-crowd regions. Hence, the desired cost of cutting a graph across it should be small too. We exploit this by having all sliding windows that contain *both* pixels l_1 and l_2 contribute to $c^{(2)}(l_1, l_2)$. Specifically, each such window increases the corresponding cost by an amount proportional to the associated log-likelihood, after it is normalized (using a sigmoid function) to the range $[-1, 1]$.

2.4 Experimental results

To evaluate the effectiveness of the proposed method, we collected a database of 100 images, half of which contain a crowd, see Fig. 6. Randomly selected 15 images from the database were hand segmented into crowd and non-crowd regions and used to train our algorithm. Training consisted of (i) interest point detection ($\approx 100,000$ in total), and (ii) clustering of the associated descriptors into 1000 SIFT words, followed by (iii) estimation of scale-invariant word densities, as in Sec. 2.2.

We used the remaining 85 images to test the performance of the proposed algorithm. A representative set of results is shown in Fig. 7, for data that contains a crowd, and in Fig. 8, for data that does not. Our method consistently demonstrated a low false negative rate (see Fig. 7), detecting crowds over a variety of viewpoints, scales, and scene types ranging from concerts and political rallies to street crowds. This result is even more impressive considering that it was achieved while maintaining a very low false positive detection rate, which is shown in red in Fig. 7. Interestingly, an examination of sources of false positive errors showed that they are dominated by "leafy" regions, which by the definition in Sec. 1 are valid crowds too. This rather unambiguously suggests that our low-level features may not be discriminative enough when used independently to differentiate between these two types of crowds. In contrast, the bottom-right image in Fig. 8 shows stacked fruit, which our algorithm correctly classifies are non-crowd.

Figure 6: Example images from our database. Note the extent of appearance variability: illumination changes are large, individuals vary in scale and pose, as does their separation.

The most immediate direction for research we intend to pursue to improve the proposed method, is that of modelling co-occurrences of SIFT words within a sliding window pyramid. We believe that this will significantly improve the discriminative power of our low-level features, thus reducing the few problematic sources of false positive detections.

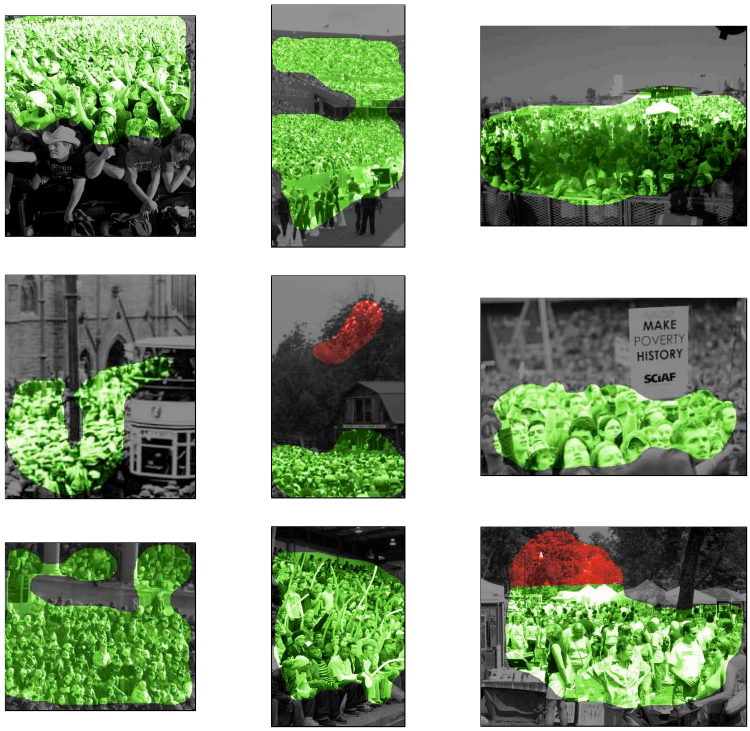


Figure 7: Segmentation results on data containing crowds: green areas show regions correctly identified as a crowd (true positives), red areas incorrectly identified as a crowd (false positives).

3 Summary and Conclusions

Our main contribution in this paper is a novel algorithm for detection of crowds of people in still images. The algorithm is appearance-based, employing a statistical, Poisson model of occurrences of quantized SIFT words across an image. The proposed method demonstrated promising results on a dataset with large variation in viewpoint, appearance of individuals in the crowd, their density and scale, and background scene type.

References

[1] G. J. Brostow and R. Cipolla. Unsupervised bayesian detection of independent motion in crowds. *In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1:594–601, 2006.

[2] D. M. Gavrila. Pedestrian detection from a moving vehicle. *In Proc. European Conference on Computer Vision (ECCV)*, 2:37–49, 2000.

[3] D. Kong, D. Gray, and H. Tao. A viewpoint invariant approach for crowd counting. *In Proc. IEEE International Conference on Pattern Recognition (ICPR)*, 3:1187–1190, 2006.

[4] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision (IJCV)*, 60(2):91–110, 2003.

Figure 8: Examples non-crowd data: all images were correctly classified as not containing crowds.

[5] V. Rabaud and S. Belongie. Counting crowded moving objects. *In Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2006.

[6] P. Reisman, O. Mano, S. Avidan, and A. Shashua. Crowd detection in video sequences. *International Symposium on Intelligent Vehicles*, pages 66–71, 2004.

[7] D. Roqueiro and V. A. Petrushin. Counting people using video cameras. *International Journal of Parallel, Emergent and Distributed Systems (IJPEDS)*, 22(3):193–209, 2007.

[8] P. Viola and M. Jones. Robust real-time face detection. *International Journal of Computer Vision (IJCV)*, 57(2):137–154, 2004.

[9] P. Viola, M. Jones, and D. Snow. Detecting pedestrians using patterns of motion appearance. *In Proc. IEEE International Conference on Computer Vision (ICCV)*, pages 734–741, 2003.

[10] B. Wu, H. Ai, C. Huang, and S. Lao. Fast rotation invariant multi-view face detection based on real adaboost. *In Proc. IEEE International Conference on Automatic Face and Gesture Recognition (FGR)*, pages 79–84, 2004.

[11] M-H. Yang, N. Ahuja, and D. Kriegman. A survey on face detection methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 24(1):34–58, 2002.