Density Estimation in Aerial Images of Large Crowds for Automatic People Counting

Christian Herrmann, Juergen Metzler

Fraunhofer Institute of Optronics, System Technologies and Image Exploitation,
Fraunhoferstrasse 1, 76131 Karlsruhe, Germany

ABSTRACT

Counting people is a common topic in the area of visual surveillance and crowd analysis. While many image-based solutions are designed to count only a few persons at the same time, like pedestrians entering a shop or watching an advertisement, there is hardly any solution for counting large crowds of several hundred persons or more. We addressed this problem previously by designing a semi-automatic system being able to count crowds consisting of hundreds or thousands of people based on aerial images of demonstrations or similar events. This system requires major user interaction to segment the image. Our principle aim is to reduce this manual interaction. To achieve this, we propose a new and automatic system. Besides counting the people in large crowds, the system yields the positions of people allowing a plausibility check by a human operator. In order to automatize the people counting system, we use crowd density estimation. The determination of crowd density is based on several features like edge intensity or spatial frequency. They indicate the density and discriminate between a crowd and other image regions like buildings, bushes or trees. We compare the performance of our automatic system to the previous semi-automatic system and to manual counting in images. By counting a test set of aerial images showing large crowds containing up to 12,000 people, the performance gain of our new system will be measured. By improving our previous system, we will increase the benefit of an image-based solution for counting people in large crowds.

Keywords: crowd density, density estimation, crowd analysis, people counting, aerial image

1. INTRODUCTION

Large public events like demonstrations always raise the question for the number of participants. Answering this question using traditional methods like tally counters leads to massive need of staff, inaccuracies and no option to check the result for plausibility. An image-based solution offers the possibility to check the counting result afterwards. In order to reduce the necessary staff, an automated solution is preferred. As output of an automated counting system, the number of people as well as their positions are required in order to be able to check the result for plausibility. We previously designed a semi-automatic system meeting these requirements[1]. However, major user interaction is included to generate a foreground segmentation. The main concept of this contribution is to use the crowd density for counting and localizing the persons in the crowd in order to avoid the manual segmentation.

Many existing image-based methods either count persons or estimate the density for image areas. Junior et al.[2] give a good summary of counting and density estimation approaches for crowds. They distinguish three categories: object-, pixel- and texture-based approaches. Object-based ones work with a person model and are mainly used for counting people. A person model could either be for the whole body[3,4] or just the head[5,6]. Pixel-based methods use special pixels like foreground[7,8], edge[7,8] or corner pixels[9] to draw conclusions about the number of persons. Technically, the number of such pixels is determined and converted to the number of people by a linear function or a pre-trained regression method. Density estimation techniques primarily use texture-based approaches. Even though texture-based methods are sometimes used to estimate the number of people[10], this category includes predominantly density estimation methods. Marana et al.[11] systematically tested several established methods from texture analysis for crowd density estimation purposes. Using the gray level dependence matrix, straight line segments, the Fourier spectrum and the fractal dimension, similar results for each method were reported. Rahmalan et al.[12] got slightly improved results for illumination variant scenarios

Airborne Intelligence, Surveillance, Reconnaissance (ISR) Systems and Applications X, edited by D. J. Henry, et al.,
Proc. of SPIE Vol. 8713, 87130V · © 2013 SPIE · CCC code: 0277-786X/13/$18 · doi: 10.1117/12.2015758

(a) (b)

Figure 1. About 13,000 people taking part in a large demonstration in Stuttgart, Germany. Two images taken from a helicopter. Courtesy of Polizeipraesidium Stuttgart[†].

using Chebyshev moments. But both approaches just estimate the crowd density for the whole image. Local density for larger crowds is estimated by Hinz[13] using a Laws-ss-filter under the assumption that a known scene model can be used to remove background patches like buildings. Combining density estimation and people counting is quite a new approach in crowd analysis used by Lempitsky et al.[14] . In this case a feature vector based on the SIFT descriptor[15] is used. A linear transformation maps the feature vector to a density. The number of persons is then estimated as the integral over the density.

Despite the variety of existing approaches, none of them fully meets our requirements. While object-based methods provide number and positions of people, they are rather limited with respect to the maximum crowd size. They are principally designed for images with less than 20 people and rather good image resolution. Existing texture-based density estimation techniques are good at basic crowd analysis. But they either do not estimate the number of people at all, or if so, do not give the positions of people. Not being able to determine the positions is also the problem with the pixel-based approaches.

We propose a novel system that gives the number and positions of people by using a density estimation. It is especially designed to process images of large crowds containing at least several hundred persons as shown in Fig. 1. We consider the same scenarios and challenges as in our previous semi-automatic system[1] . Briefly, this means single images with arbitrary crowd densities, differing person size and large numbers of people. Usually the scene is outdoors and not known previously. We made the assumption for the input image to be an approximate vertical view of the crowd ensuring a homogeneous person size of the crowd in the image.

2. SYSTEM DESIGN

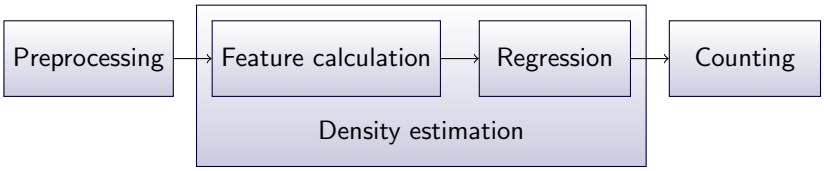
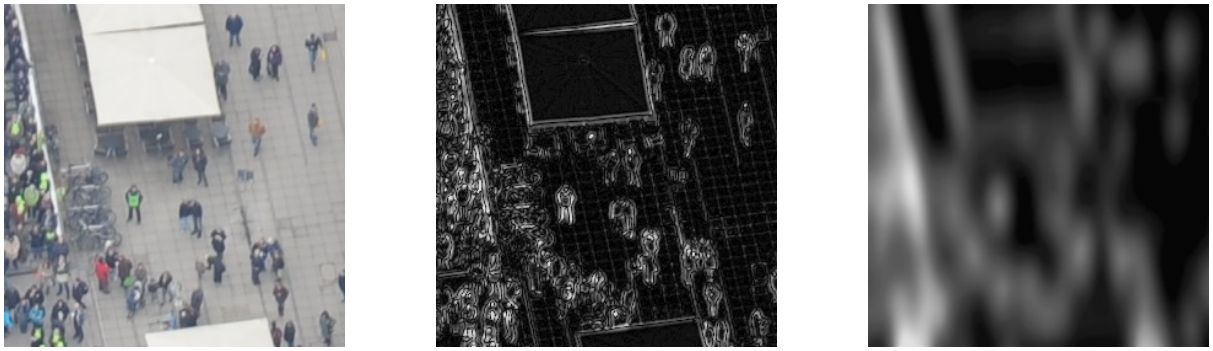


Figure 2. Steps in the counting system.

We use a system design with three major steps as shown in Fig. 2. Basically, it consists of a texture-based density estimation step and an object-based step to count and localize the persons. We avoid the limitations

[†]http://org.polizei-bwl.de/ppstuttgart/PublishingImages/s21/Luftaufnahmen/DSC_8621.JPG and http://org.polizei-bwl.de/ppstuttgart/PublishingImages/s21/Luftaufnahmen/DSC_8612.JPG



(a) (b) (c)

Figure 3. (a) Input image, (b) high-pass filtered image I_{hp} and (c) high-pass feature f_{hp}.

of object-based approaches in crowd size by using a simple person model. It detects the persons in the crowd density map instead of the original image. Additionally, a preprocessing step at the beginning prepares the input images. Preprocessing includes conversion to grayscale and person size normalization. As the person size varies heavily from image to image we need to eliminate scale effects. To achieve this, we scale each image with different factors for width and height according to the specific person size. After the preprocessing step, the person size is fixed to a predefined size. We found a normalized person size of 16 $\times$ 16 pixel to be the best trade-off between performance and computational effort. The preprocessed, normalized image I_n serves as input image to the following steps.

3. DENSITY ESTIMATION

The main contribution of this work is the estimation of the crowd density. The basic idea is a feature-based approach as it is used by Lempitsky et al.[14] . Out of a set of local features, the crowd density is estimated for each pixel by some regression method. Thus, for each pixel a set of features which indicates the density needs to be determined. The challenges in finding useful features are on the one hand the discrimination between persons and background and on the other hand the quantification of crowd density itself if persons are residing at the relevant spot. As we said before, all calculations of features will be performed on the normalized input image. However, for the sake of visibility and clarity, the aspect ratio of the example images in the shown figures is not normalized.

3.1 Features

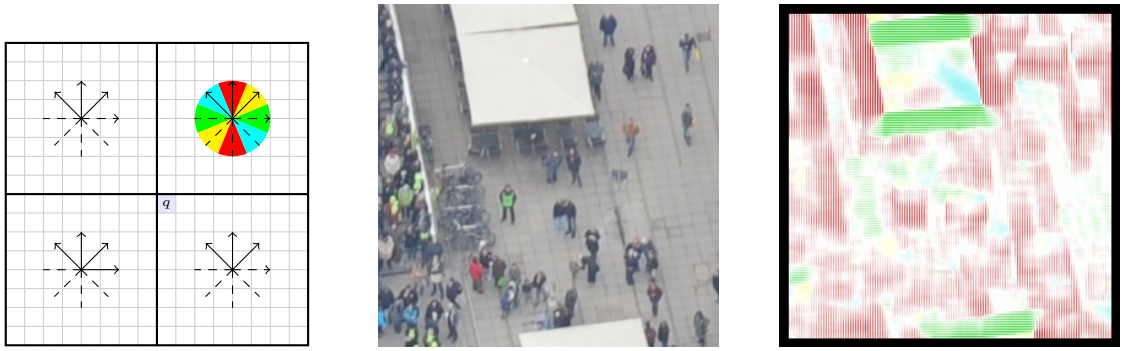
The first feature is based on the high-pass filtered image. Due to people's different clothing, increasing crowd density leads to a denser color pattern in the image. In contrast to background areas like streets or lawn, these color patterns have a higher spatial frequency from a signal theoretic point of view. By applying a high-pass filter on the normalized input image I_n we get:

$$I_{hp} = HP\{I_n\}\,. \tag{1}$$

The filtered image I_{hp} mostly indicates edges as they induce high spatial frequencies (Fig. 3). But crowd density should not be rated highest at the edge of a person, but at its center. To achieve this, we apply a low-pass filter on the already filtered image I_{hp}. This leads to an appropriate feature f_{hp} for each position $\boldsymbol{x} = (x, y)^T$ in the image (we indicate vectors with bold and scalar values with regular symbols):

$$f_{hp}(\boldsymbol{x}) = TP\{I_{hp}\}(\boldsymbol{x})\,. \tag{2}$$

Fig. 3 shows the output of the several steps in the processing chain. The example scene includes a large variety of difficulties our system has to cope with: structured background, stairs with persons on the left side, umbrellas, bikes, chairs and variable crowd density. However, very inhomogeneous backgrounds are not distinguishable from crowds with this feature.



(a) (b) (c)

Figure 4. (a) 16 × 16 neighborhood of a pixel q with the four subareas. The upper right subarea shows the colors used to indicate the different edge orientations. (b) Input image. (c) Colored illustration of the edge orientation feature $\boldsymbol{f}_e$. The color/pattern indicates the edge orientation $f_{es,1}$, the intensity indicates the strength $f_{es,2}$.

Lempitsky et al.[14] mainly use the SIFT descriptor[15] for density estimation. The descriptor describes the neighborhood N_q of a pixel q with a 128-dimensional vector. But using this many dimensions is disadvantageous in our context, as the descriptor is designed for unique identification of image points. If not the unique identification of a point is necessary, but just the determination of the crowd density, the high dimensionality does not make sense. The reduction to 16 dimensions uses the following structure: the descriptor covers an area of a normalized person with 16 × 16 pixels and is divided in 2 × 2 subareas. Each subarea contains 8 × 8 pixels and is analyzed for edge orientations. Collecting the edge orientations and intensities in histograms $\boldsymbol{h}_i,\ i = 1, ..., 4$ using 4 orientation-bins (horizontal, vertical and both diagonal directions) leads to 4 subareas × 4 orientations = 16 dimensions (see Fig. 4a). The feature vector $\boldsymbol{f}_e$ is based on the descriptor and contains the bin values of all 4 histograms $\boldsymbol{h}_i = (h_{i,1}, h_{i,2}, h_{i,3}, h_{i,4})^T$:

$$\boldsymbol{f}_e = \begin{pmatrix} h_{1,1} \\ h_{1,2} \\ \vdots \\ h_{4,4} \end{pmatrix} . \tag{3}$$

We use this edge orientation feature especially for discrimination between people and artificial background. Buildings or streets lead to long and strong edges while crowds generate more randomly oriented edges. For display purposes we use the simplification $\boldsymbol{f}_{es} = (f_{es,1}, f_{es,2})^T$ of $\boldsymbol{f}_e$:

$$f_{es,1} = \arg\max_i h_{all,i} \quad \text{and} \quad f_{es,2} = \frac{h_{all,f_{es,1}}}{\sum_i h_{all,i}} \quad \text{, with} \quad \boldsymbol{h}_{all} = \sum_{i=1}^{4} \boldsymbol{h}_i \,. \tag{4}$$

The simplification $\boldsymbol{f}_{es}$ contains the main edge orientation $f_{es,1}$ of the whole descriptor area and the relative intensity $f_{es,2}$ of this orientation. Fig. 4c shows that the feature distinguishes between the strong edges of the umbrellas and the softer and more random ones in the crowd as expected.

Measures based on Fourier spectrum may serve as further feature. The spatial frequency has already proved to be useful for estimating the person density[11] . A denser crowd leads to a finer texture in the image. As a result, the energy in the Fourier spectrum moves to higher frequencies. A spectrum analysis should particularly be able to discriminate between a crowd and optically similar structures like trees or bushes. This discrimination is not trivial at all, as both situations produce irregular statistical textures in the image (Fig. 5). Inspired by Marana et al.[11] two characteristics of the frequency spectrum are used: distribution over frequency $\boldsymbol{f}_{freq}$ and isotropy $\boldsymbol{f}_{iso}$. In contrast to the approach from Marana et al., we need a local estimation for crowd density instead of a global one for the whole image region. By windowing a 16 × 16 neighborhood N_q of each pixel q with $w(r) = e^{-lr^2}$, we are able to perform a local Fourier analysis for each pixel. As we use a Gaussian function as window function this leads to a Gabor transform of the whole image. The radius r of the window function denotes the distance of

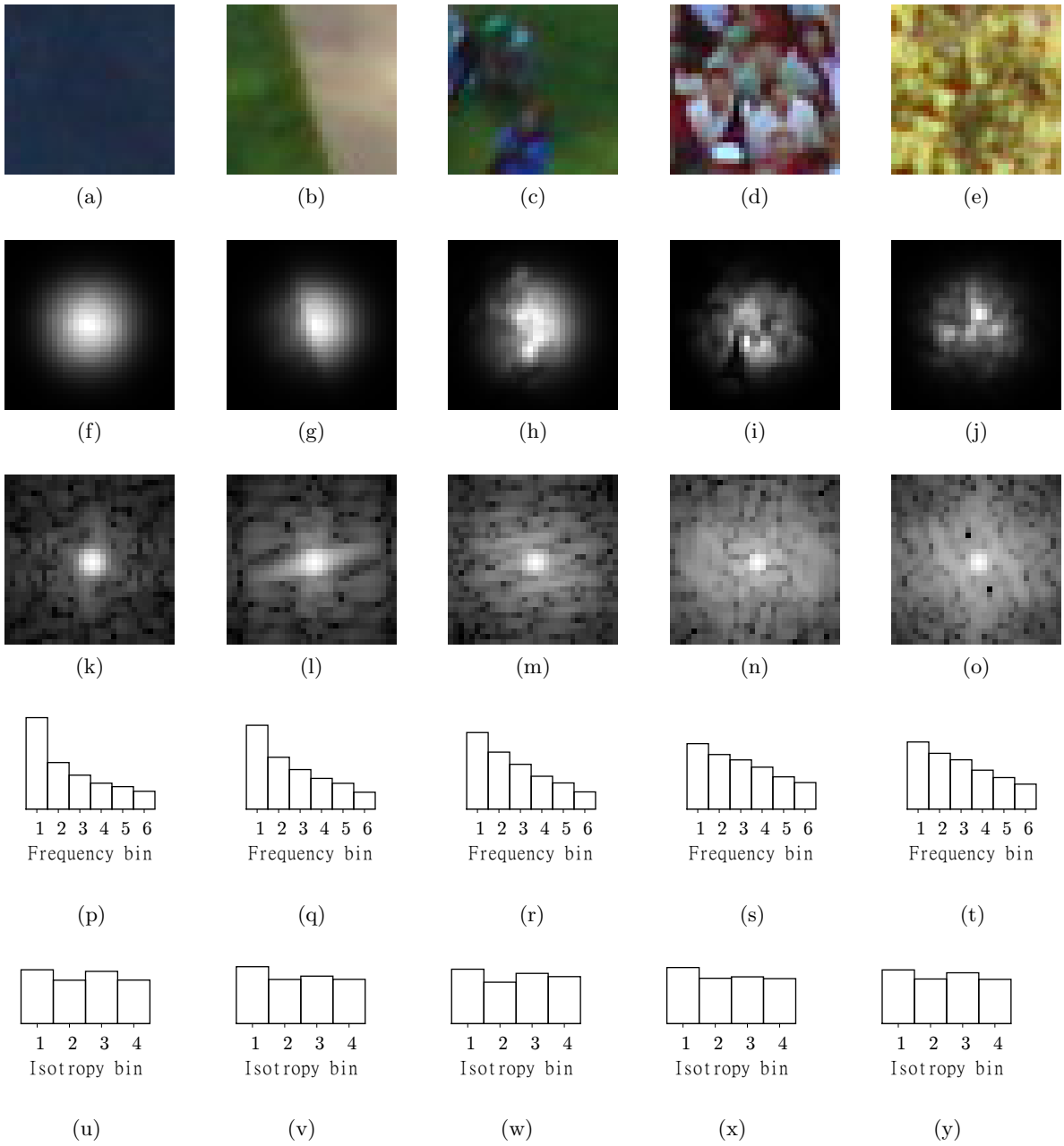
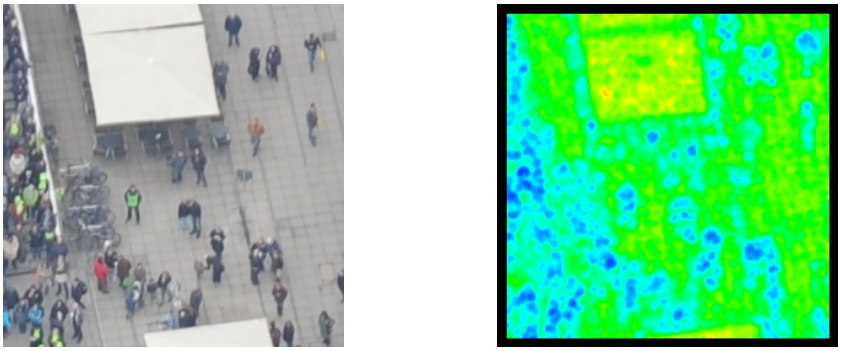


Figure 5. Selection of scenarios and their corresponding Fourier features: (a) uniform background, (b) roadside, (c) loose crowd, (d) dense crowd and (e) dense bushes. (f)-(j) Windowed and grayscale regions, (k)-(o) Fourier spectrum, (p)-(t) frequency feature $\boldsymbol{f}_{freq}$ and (u)-(y) isotropy feature $\boldsymbol{f}_{iso}$.



(a) (b)

Figure 6. (a) Input image and (b) colored illustration of the feature $\boldsymbol{f}_{freq}$ by the weighted mean μ. Warm/Bright colors (red, yellow) denote low frequencies, cold/dark colors (cyan, blue) high frequencies.

a neighborhood pixel to the central pixel of the neighborhood, while l denotes a constant parameter regulating the window intensity. After locally performing the Fourier transform $F\{\overline{N}_q\}$ on the windowed neighborhood $\overline{N}_q$, the magnitude spectrum $S_q = |F\{\overline{N}_q\}|$ is used as it contains the information about the periodical structures. By transformation of the local spectrum to the polar coordinate system, the features $\boldsymbol{f}_{freq}$ and $\boldsymbol{f}_{iso}$ result from the projections:

$$S_q(r) = \int_0^{\pi} S_q(r,\varphi)\mathrm{d}\varphi\,, \quad S_q(\varphi) = \int_0^{R} S_q(r,\varphi)\mathrm{d}r\,, \tag{5}$$

where R denotes the maximum radius. By division in n frequency and m orientation bins the k-th components of the features are:

$$f_{freq,k} = \int_{(k-1)\frac{R}{n}}^{k\frac{R}{n}} S_q(r)\mathrm{d}r\,, \quad f_{iso,k} = \int_{(k-1)\frac{\pi}{m}}^{k\frac{\pi}{m}} S_q(\varphi)\mathrm{d}\varphi\,. \tag{6}$$

We divided the frequency in $n = 6$ and the orientation in $m = 4$ bins. Fig. 5 shows examples for the features in different scenarios. An illustration of $\boldsymbol{f}_{freq}$ for our example scene is shown in Fig. 6b. We used the weighted mean μ of the bin indices for a colored representation:

$$\mu = \frac{1}{n}\sum_{k=1}^{n} f_{freq,k} \cdot k\,. \tag{7}$$

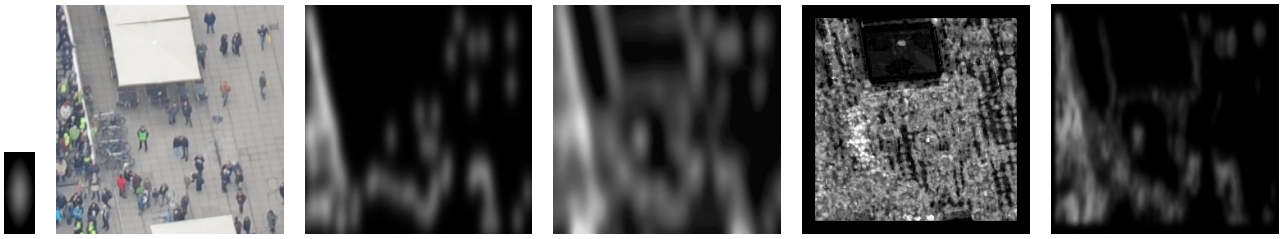
3.2 Regression

Based on the defined features we can generate a density map D, where $D : Q \to [0,\infty)$ and Q is the set of all image pixels. If no persons are around, the density is zero. Otherwise the value of D increases with the crowd density. For specifying the ground truth density D_{gt} we use kernel density estimation. This method creates a continuous density out of point annotations. Each person in the training set of images needs to be annotated with its center point $\boldsymbol{p}_i$, then the density is given by

$$D_{gt}(\boldsymbol{x}) = \sum_{i=1}^{n_t} \varphi(\boldsymbol{x} - \boldsymbol{p}_i)\,, \tag{8}$$

where φ is a kernel function and n_t the number of persons. We use a two-dimensional Gaussian function as kernel function

$$\varphi_{w,h}(\boldsymbol{x}) = G_{w,h}(\boldsymbol{x}) = \frac{1}{2\pi\sigma(w)\sigma(h)}\; e^{-\frac{1}{2}\left(\frac{x^2}{\sigma(w)^2}+\frac{y^2}{\sigma(h)^2}\right)}\,. \tag{9}$$



(a) (b) (c) (d) (e) (f)

Figure 7. (a) Kernel function - two-dimensional Gaussian function, (b) original image, (c) ground truth density D_{gt}, (d) high-pass feature f_{hp}, (e) regression result D_{reg} - calculated for all pixels for illustration purposes and (f) final density map D.

The size of the Gaussian function is given by $\sigma(w)$ and $\sigma(h)$, which depend on the person size $\boldsymbol{s} = (w, h)^T$ in the image. We used $\sigma(l) = \frac{3}{5}\, l + \frac{1}{2}$. The resulting ground truth density D_{gt} can be seen in Fig. 7c.

The ordinary solution to estimate D out of the features would be to put all of them into a combined feature vector $\tilde{\boldsymbol{f}} = (f_{hp}^T, \boldsymbol{f}_e^T, \boldsymbol{f}_{freq}^T, \boldsymbol{f}_{iso}^T)^T$ and train a regression method with the help of the ground truth density.

Although this would work, there is a more efficient way to perform this task: by applying a threshold to the high-pass feature, the remaining features need to be calculated only for a reduces number of positions. If we compare f_{hp} to the ground truth density D_{gt}, we see that they are very similar at positions with persons (Fig. 7c and 7d). But as there are also further positions without persons where f_{hp} shows values bigger than zero, we can just use the assumption

$$f_{hp}(\boldsymbol{x}) \leq T_{f_{hp}} \Rightarrow D_{gt}(\boldsymbol{x}) = 0\,, \tag{10}$$

where $T_{f_{hp}}$ denotes a threshold. By using the fast to calculate high-pass feature f_{hp} and a threshold $T_{f_{hp}}$, only the positions in the image where f_{hp} exceeds $T_{f_{hp}}$ need further inspection. This reduces the computational effort for the more complex features and supports the training of the regression method, as irrelevant data does not need to be considered. For estimating the density D, the threshold $T_{f_{hp}}$ is applied:

$$\overline{f}_{hp}(\boldsymbol{x}) = \begin{cases} f_{hp}(\boldsymbol{x}) & ,\ \text{if } f_{hp}(\boldsymbol{x}) > T_{f_{hp}} \\ 0 & ,\ \text{otherwise} \end{cases}\,. \tag{11}$$

For all positions $\boldsymbol{u} \in U$ with

$$U = \left\{ \boldsymbol{u} \,|\, \overline{f}_{hp}(\boldsymbol{u}) \neq 0 \right\}\,, \tag{12}$$

the further features $\boldsymbol{f}_e, \boldsymbol{f}_{freq}, \boldsymbol{f}_{iso}$ or a reasonable subset of them are combined to a feature vector $\boldsymbol{f}(\boldsymbol{u})$. Leaving one feature out might be useful in the case of $\boldsymbol{f}_e$ or $\boldsymbol{f}_{iso}$, as they both contain similar information. For regression we use gradient boosted trees to get the density $D_{reg}(\boldsymbol{u})$ (Fig. 7e). The final density D results the following way:

$$D(\boldsymbol{x}) = \begin{cases} \overline{f}_{hp}(\boldsymbol{x}) \cdot D_{reg}(\boldsymbol{x}) & ,\ \text{if } \boldsymbol{x} \in U \\ 0 & ,\ \text{otherwise} \end{cases}\,. \tag{13}$$

A resulting density map D can be seen in Fig. 7f.

4. PEOPLE COUNTING

For counting and locating the people, we use a similar approach as in our previous system[1]. We changed the rectangular person model to the Gaussian model used for ground truth generation in the section before. For the iterative subtraction of the person model $P = G_{w,h}(\boldsymbol{x})$ from the density D we just need a measure $M_P(D, \boldsymbol{x}_0)$ that indicates how well the model fits at a specific position $\boldsymbol{x}_0 = (x_0, y_0)^T$. Defining the basic measure $\overline{M}_P(D, \boldsymbol{x}_0)$ as integral over the absolute difference between an adapted person model P and the density D has the advantage

(a) (b) (c) (d)

Figure 8. (a) Calculated density D, (b) detected persons based on density D (white circles), (c) illustration of the 66 counted person positions in the input image and (d) illustration of the manually annotated 72 persons.

of a certain tolerance towards errors in the density D:

$$\overline{M}_P(D, \boldsymbol{x}_0) = \int_{x_0-\frac{w}{2}}^{x_0+\frac{w}{2}} \int_{y_0-\frac{h}{2}}^{y_0+\frac{h}{2}} |D(\boldsymbol{x}) - l_{\boldsymbol{x}_0} \cdot P(\boldsymbol{x} - \boldsymbol{x}_0)| \, \mathrm{d}x\mathrm{d}y \,. \tag{14}$$

The scaling factor $l_{\boldsymbol{x}_0}$ is chosen in a way that the central value $P(\mathbf{0})$ of the model is adjusted to the density $D(\boldsymbol{x}_0)$:

$$l_{\boldsymbol{x}_0} = \frac{D(\boldsymbol{x}_0)}{P(\mathbf{0})} \,. \tag{15}$$

Further T_1 and T_2 are two thresholds with $T_1 < l_{\boldsymbol{x}_0} < T_2$. If this condition is not met, the value $\overline{M}_P(D, \boldsymbol{x}_0)$ is not valid. The lower threshold avoids a best possible result if the density at $\boldsymbol{x}_0$ and in its neighborhood is 0. The upper threshold avoids wrong person detections at positions with a dense crowd where multiple persons occlude each other. Finally, the measure is defined as:

$$M_P(D, \boldsymbol{x}_0) = \begin{cases} \overline{M}_P(D, \boldsymbol{x}_0) & , \text{ if } T_1 < l_{\boldsymbol{x}_0} < T_2 \\ \infty & , \text{ otherwise} \end{cases} \,. \tag{16}$$

In this way $M_P(D, \boldsymbol{x}_0)$ measures the distance between P and D. The best accordance for P and D is determined by:

$$\boldsymbol{x}_b = arg \min_{\boldsymbol{x}_0} M_P(D, \boldsymbol{x}_0) \,. \tag{17}$$

Using this approach leads to the counting result shown in Fig. 8 for our example scene.

5. RESULTS

We evaluate our system in two steps. First the performance of the density estimation and afterwards the counting performance will be presented. The test data set contains 16 images, 14 of them are annotated with the person positions. The number of people ranges from 2 to 12,000 in the images and most of them contain about one hundred people.

5.1 Density estimation

The 14 annotated images consist of about 2 million pixel together, for which a density estimation is possible. Some border pixels drop out as most features work with a 16 × 16 region that has to lie on the inside of the image. A random set of one percent of the pixels is used for training, so that about 20,000 feature vectors are available for training. All other feature vectors are used as test data. The quality of the estimated density is

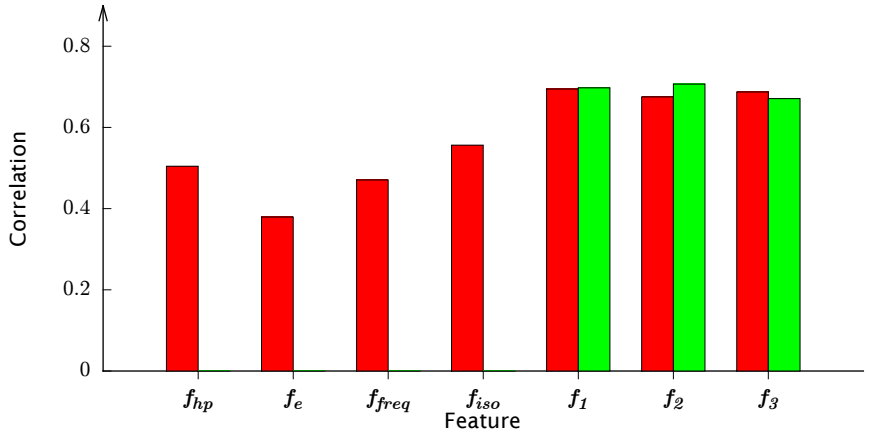


Figure 9. Correlation coefficient r between ground truth D_{gt} and estimated density D using the respective feature vector. Colors indicate different methods: ordinary regression (red/dark gray) and proposed thresholding method (green/light gray). Combined feature vectors: $\boldsymbol{f}_1 = (\boldsymbol{f}_e^T, \boldsymbol{f}_{freq}^T, \boldsymbol{f}_{iso}^T)^T$, $\boldsymbol{f}_2 = (\boldsymbol{f}_e^T, \boldsymbol{f}_{freq}^T)^T$ and $\boldsymbol{f}_3 = (\boldsymbol{f}_{freq}^T, \boldsymbol{f}_{iso}^T)^T$.

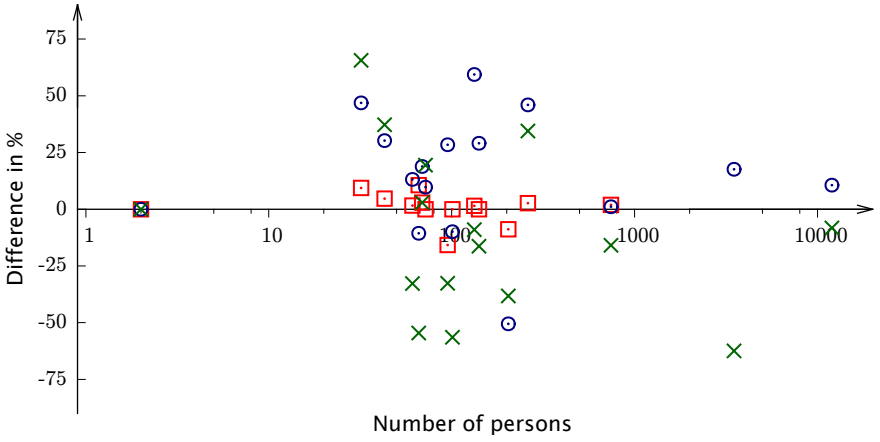


Figure 10. Counting difference d in dependency of the number of persons n_a in the image. Compared between manual (boxes), semi-automatic counting (circles) and density-based counting (crosses).

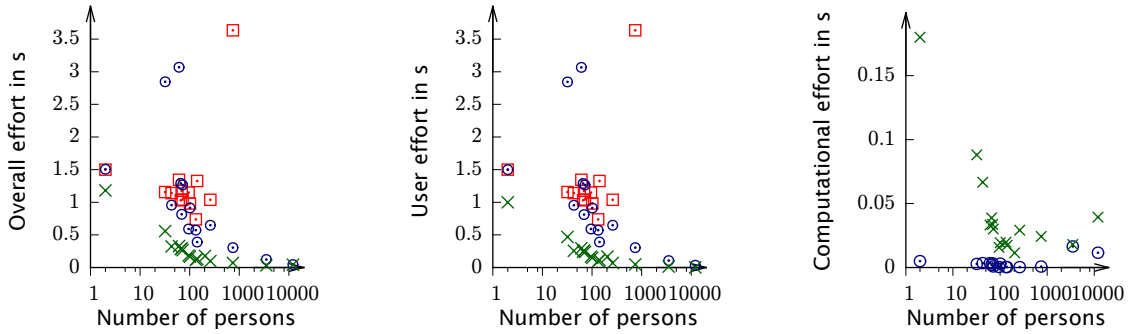
measured by the Pearson's correlation coefficient r between the ground truth density D_{gt} and the estimated density D for all N test samples :

$$r = \frac{\sum_{i=1}^{N} (D_{gt,i} - \overline{D}_{gt})(D_i - \overline{D})}{\sqrt{\sum_{i=1}^{N} (D_{gt,i} - \overline{D}_{gt})^2 \sum_{i=1}^{N} (D_i - \overline{D})^2}} \,. \tag{18}$$

$\overline{D}$ and $\overline{D}_{gt}$ are the means of D and D_{gt}. Fig. 9 shows the results for different feature vectors and a comparison of the ordinary regression to our proposed thresholding method based on the high-pass feature. Besides testing each feature for itself, three different combinations of features are tested. In addition to the full feature vector $\boldsymbol{f}_1$, we test leaving out each one of the orientation based features $\boldsymbol{f}_e$ and $\boldsymbol{f}_{iso}$. The results show that leaving one of them out does not have a major impact on the performance. Not using $\boldsymbol{f}_{iso}$ even increases the performance by a small amount. Furthermore, our proposed thresholding method reaches the performance of the ordinary regression, and it is the preferred method as it reduces the computational effort by a large amount. Hence, we

Table 1. Mean computation times for the individual processing steps. Given as mean time to count one person.

Processing step	Time per person in ms	
	ordinary regression	thresholding method
Normalization	0,07	0,07
High-pass feature f_{hp}	0,03	0,03
Edge orientation feature $\boldsymbol{f}_e$	5,47	2,90
Fourier features $\boldsymbol{f}_{freq}$, $\boldsymbol{f}_{iso}$	28,62	14,98
Regression	5,23	2,88
Counting	15,08	15,08
Overall	54,49	35,94



(a) (b) (c)

Figure 11. Effort per person in dependency of the number of persons n_a in the image. Compared between manual counting (boxes), semi-automatic counting (circles) and density-based counting (crosses): (a) overall effort $\bar{t}_o = \bar{t}_u + \bar{t}_c$, (b) user effort $\bar{t}_u$ and (c) computational effort $\bar{t}_c$.

use $\boldsymbol{f}_2 = (\boldsymbol{f}_e^T, \boldsymbol{f}_{freq}^T)^T$ as feature combination with the thresholding method for the further tests.

5.2 Counting

The counting performance is evaluated with respect to user effort, computational effort and accuracy. The system is compared to our previous semi-automatic system and a manual counting of the people in each category. We performed the same test scenarios as for our previous system, so that the results can be compared. For this test, we use the whole dataset of 16 images. If we denote the actual number of people in the image as n_a and the counted number as n_c, the accuracy can be given as relative difference d:

$$d = \frac{n_c - n_a}{n_a} \cdot 100\% \,. \tag{19}$$

The algebraic sign indicates if the number of persons is over- (positive sign) or underestimated (negative sign). Our proposed system reaches a mean absolute difference of 29 percent. The previous system achieved 24 percent, while manual counting with 4 percent difference yields the best results. Fig. 10 shows the detailed distribution over the test set. Although we observe a slight decrease in counting performance, our new system has the advantage of heavily reduced user effort. Using the user effort t_u and the computational effort t_c, normalized versions are defined:

$$\bar{t}_u = \frac{t_u}{n_a} \;,\; \bar{t}_c = \frac{t_c}{n_a} \,. \tag{20}$$

The normalization with respect to the actual number of persons n_a allows to compare times between different images. The time measurement results in Fig. 11 clearly show the reduced effort. The new system is considerably

faster for all images. Overall, an average reduction by 75 percent compared to the previous system can be achieved. We have some remaining user effort as we account for manual person size determination in the normalization step. Using camera calibration information, if available, would clearly eliminate this part.

The computational effort of our new system is distributed as shown in Table 1. Normalization and high-pass feature hardly contribute to the entire effort. Most of the time is used for the Fourier-based features. Their computation time can not be separated in a reasonable way as the Fourier transformation takes most of the time and can be used for both. Using the threshold approach leads to half of the original effort in the feature calculation steps.

6. CONCLUSION

By using a density estimation approach, we reached a considerable decrease in processing effort for counting and locating people in crowds. Removing the manual segmentation step is the beneficial key to this solution. The feature based method allows a local density estimation which implicitly contains a segmentation of the image. This makes the previously manual segmentation step unnecessary. The usage of the person model for counting has the benefit of extracting the positions of the people and not only the number. This makes it superior to a simple integration over the density. The counting accuracy remains with 29 percent difference at almost the same level as before, while we achieve a four times faster processing speed.

The modular system design enables further applications which use just the density estimation. It might be useful to detect dangerous crowd densities and prevent damage to persons by enabling appropriate countermeasures. An adaption to infrared images seems possible with a minor effort of retraining and possibly other composition of the feature vector. Although new features might need to be designed to get useful results for different target objects, the extension to other counting scenarios instead of persons in a crowd seems possible as well.

REFERENCES

[1] Herrmann, C., Metzler, J., and Willersinn, D., "Semi-automatic people counting in aerial images of large crowds," in [*SPIE Security + Defence*], International Society for Optics and Photonics (2012).

[2] Junior, J., Musse, S., and Jung, C., "Crowd analysis using computer vision techniques," *IEEE Signal Processing Magazine* **27**(5), 66–77 (2010).

[3] Zhao, T. and Nevatia, R., "Bayesian human segmentation in crowded situations," in [*IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings*], **2**, 459–466, IEEE (2003).

[4] Schofield, A., Stonham, T., and Mehta, P., "Automated people counting to aid lift control," *Automation in Construction* **6**(5-6), 437–445 (1997).

[5] Lin, S., Chen, J., and Chao, H., "Estimation of number of people in crowded scenes using perspective transformation," *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans* **31**(6), 645–654 (2001).

[6] Zhang, X. and Sexton, G., "Automatic human head location for pedestrian counting," in [*Sixth International Conference on Image Processing and Its Applications, 1997*], **2**, 535–540, IET (1997).

[7] Velastin, S., Yin, J., Davies, A., Vicencio-Silva, M., Allsop, R., and Penn, A., "Automated measurement of crowd density and motion using image processing," in [*Seventh International Conference on Road Traffic Monitoring and Control, 1994*], 127–132, IET (1994).

[8] Davies, A., Yin, J., and Velastin, S., "Crowd monitoring using image processing," *Electronics & Communication Engineering Journal* **7**(1), 37–47 (1995).

[9] Conte, D., Foggia, P., Percannella, G., Tufano, F., and Vento, M., "A method for counting people in crowded scenes," in [*Advanced Video and Signal Based Surveillance (AVSS), 2010 Seventh IEEE International Conference on*], 225–232, IEEE (2010).

[10] Wu, X., Liang, G., Lee, K., and Xu, Y., "Crowd density estimation using texture analysis and learning," in [*IEEE International Conference on Robotics and Biomimetics, 2006. ROBIO'06.*], 214–219, IEEE (2006).

[11] Marana, A., da Costa, L., Lotufo, R., and Velastin, S., "On the efficiency of texture analysis for crowd monitoring," *Computer Graphics, Image Processing and Vision, Washington, DC* , 354ff. (1998).

[12] Rahmalan, H., Nixon, M., and Carter, J., "On crowd density estimation for surveillance," in [*Crime and Security, 2006. The Institution of Engineering and Technology Conference on*], 540–545, IET (2006).

[13] Hinz, S., "Density and Motion Estimation of People in Crowded Environments Based on Aerial Image Sequences," *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 38(1-4-7/W5), on CD* (2009).

[14] Lempitsky, V. and Zisserman, A., "Learning to count objects in images," *Advances in Neural Information Processing Systems* (2010).

[15] Lowe, D., "Distinctive image features from scale-invariant keypoints," *International journal of computer vision* **60**(2), 91–110 (2004).