Estimation of crowd density from UAVs images based on corner detection procedures and clustering analysis

Ali Almagbile
Department of Geography, Yarmouk University, Irbid, Jordan

ABSTRACT
With rapid developments in platforms and sensors technology in terms of digital cameras and video recordings, crowd monitoring has taken a considerable attentions in many disciplines such as psychology, sociology, engineering, and computer vision. This is due to the fact that, monitoring of the crowd is necessary to enhance safety and controllable movements to minimize the risk particularly in highly crowded incidents (e.g. sports). One of the platforms that have been extensively employed in crowd monitoring is unmanned aerial vehicles (UAVs), because UAVs have the capability to acquiring fast, low costs, high-resolution and real-time images over crowd areas. In addition, geo-referenced images can also be provided through integration of on-board positioning sensors (e.g. GPS/IMU) with vision sensors (digital cameras and laser scanner). In this paper, a new testing procedure based on feature from accelerated segment test (FAST) algorithms is introduced to detect the crowd features from UAV images taken from different camera orientations and positions. The proposed test started with converting a circle of 16 pixels surrounding the center pixel into a vector and sorting it in ascending/descending order. A single pixel which takes the ranking number 9 (for FAST-9) or 12 (for FAST-12) was then compared with the center pixel. Accuracy assessment in terms of completeness and correctness was used to assess the performance of the new testing procedure before and after filtering the crowd features. The results show that the proposed algorithms are able to extract crowd features from different UAV images. Overall, the values of Completeness range from 55 to 70 % whereas the range of correctness values was 91 to 94 %.

1. Introduction
Crowd analysis is considered a hot research topic in various fields such as psychology, sociology, engineering and computer vision (Zhan et al. 2008; Burkert and Butenuth 2011). For example, psychologists and sociologists focus on exploring individual behavior to enhance safety services for people, particularly in overcrowded areas. This can be achieved through understanding people interaction within a specific area. Many universities and research agencies such as Defence Advances Research Projects Agency (DARPA) and the British Engineering and Physical Sciences Research Council (EPSRC) The EU funded projects PRISMATICA and ADVISOR (Zhan et al. 2008) have already focused on detecting, counting, estimating and tracking moving people using either images or video recordings.

Originally, crowd monitoring and management required constructing a framework that provides surveillance and crowd control to respond to the event situation. This framework includes three consecutive steps namely sensing, alerting and action (SAA). Sensing is capturing images of crowd areas using a camera (digital/infrared/multi-spectral) mounted on-board of a moving platform, such as unmanned aerial vehicle (UAV). Furthermore, these images can be geo-referenced images particularly when using positioning sensors such as GPS, IMU or integrated GPS/IMU systems. In recent years, UAVs have been utilized to perform such tasks. This is due to their capability to acquire fast, low cost and real-time images (Lambers et al. 2007; Patterson and Brescia 2008; Nagai et al. 2009).

The captured images of the crowd area are then transmitted via communication infrastructure (wireless communication) for further processing. In the second step, images are segmented and classified to estimate crowd density. This means extracting crowd features using well-known image segmentation techniques such as background removal (Davies, Yin, and Velastin 2005), image processing and pattern recognition (Marana et al. 1998), information fusion (Velastin et al. 1994) and feature extraction points ( Conte et al. 2010).

The final step is to take an action based on the information provided in the image segmentation step. This can be achieved through providing instructions for individuals to follow the best directions that guarantee safe exit and controllable movements. Thus, the risk which is associated with people movements in
overcrowded areas can be monitored and minimized. Therefore, image segmentation step is necessary to be precisely performed. Some image segmentation technique may falsely detect some features and consider them as crowd features. Other techniques however, may miss detect the crowd features and thus many crowd features pass without detection. As a result, when the images are precisely segmented, real crowd density can be estimated.

In some places and events, crowd monitoring is necessary to provide safe and peaceful movement to minimize the risk in some incidents. This will enhance decision makers using accurate information to guide people in the field. This paper provides a new testing procedure based on the feature from accelerated segment test (FAST) algorithm for estimation crowd density from UAV images. The proposed test is based on improving the FAST method in terms of efficient processing time with high accuracy results of crowd feature detection. In addition, crowd features are then processed through filtering procedures before mapping the levels of crowd density.

The structure of the paper is as follows: related research work is presented in the second section; the following section presents the feature extraction procedures including the FAST algorithm, the proposed testing procedure, filtering and mapping the crowd features. Then test and results are provided followed by discussion and conclusions in the last section.

2. Related work

Different methodologies have been proposed for crowd density estimation. Background removal (Yin, Velastin, and Davies 1995; Davies, Yin, and Velastin 1995) is an instance of these research works which are employed on a reference image with only background to segment image pixels to either pedestrians or background. In other words, it is based on subtracting foreground from background pixels using statistical pixel information. In spite of achieving promising results by this method, it works best only for low-density crowds.

Image processing and pattern recognition were conducted by Marana et al. (1998). In this study, the authors used texture analysis for estimating crowd density. The basic idea of this method was to define the levels of crowd density as a response to the texture features. When the pattern of texture is fine, high crowd density occurs whereas coarse patterns mean low crowd density. The study showed that high crowd density was effectively estimated. However, the accuracy was degraded in the case of low crowd density. Similar procedures were conducted by Jiang et al. (2014) to classify the density into crowd regions and extremely crowded regions. In this approach, a combination of pixel statistical methods and texture analysis was performed. Then the number of people was calculated according to the corresponding fitting straight line which is based on the linear regression method. The study, however, has not provided sufficient results to prove the effectiveness of the proposed method. Wang et al. (2013) proposed a new approach for crowd density estimation through using a new texture feature called Tamura. Crowd features were extracted based on the grey level co-occurrence matrix (GLCM). Then principal component analysis (PCA) was employed to reduce the dimension of the feature vector before using Support Vector Machine (SVM) for crowd density estimation. This approach, however, was only implemented for close range images and has not shown whether it suits airborne images or not.

The information fusion method has been used to count people in crowded areas using image sensors (Yang, Gonzalez-Banos, and Guibas 2003). In this approach, the locations of people were determined using silhouettes extracted by each sensor. Velastin et al. (1994) has also employed Kalman filtering to integrate edge-based techniques and background removal for estimate crowd density. However, when sensor information is used, huge computation procedures are required and it may fail if the Kalman filter covariance matrices are not properly initiated.

Regarding crowd feature extraction procedures, corner detection approaches were used to segment the crowd features from background. Examples of these techniques are FAST and Harris corner detector (Harris and Stephens 1988; Xu et al. 2014). Butenuth et al. (2011) used simulated pedestrians for detecting and tracking pedestrians and dense regions using aerial image sequences. Crowd features were extracted using FAST and then used as input observations in Kernel density estimation to generate a map of crowd density. Graph-based event detection and hidden Markov model (HMM) were then employed to estimate the motion of pedestrians. Likewise, Sirmacek and Reina rtz (2011) employed a FAST feature detector approach and proposed an automated detection technique to detect blob-like and corner-like image structures. Then crowd density from aerial images was estimated. Whereas, Fradi and Dugelay (2013) used the FAST method to detect crowd features and added one more step for feature selection procedures to reject those local features which did not belong to crowd. However, these investigations neither discussed the type of FAST version (FAST-9, FAST-12 and FASTER) that truly detects the crowd features nor did they develop a computation procedure that improved repeatability of feature points and computational efficiency.

3. Crowd feature extraction

Local feature extraction methods have been used to extract crowd features to map the levels of crowd
density. Among these methods is the FAST method (Rosten and Drummond 2005; Butenuth et al. 2011). The FAST method, which was particularly developed for corner detection, can distinguish the value of each pixel from neighborhood pixels. The principle of the FAST method depends on a circle of 16 pixels as a test mask around a candidate corner ($I_p$) (the center pixel). This candidate corner can be a corner if at least $N$ connected pixels on the circle are brighter or darker than a threshold determined by the center pixel value (Rosten, Porter, and Drummond 2010).

Any pixel in the circle (1...16) can be one of three states (Mair et al. 2010; Biadgie and Sohn 2014):

$$S_{p-x} = \begin{cases} d, & I_{p-x} \leq I_p - t \\ s, & I_p - t < I_{p-x} < I_p + t \\ b, & I_{p-x} \geq I_p + t \end{cases}$$

where $d$ means darker; $s$ means similar; $b$ means brighter; $t$ denotes threshold, $P_d$ is darker pixels than the intensity of center pixel $I_p$, $P_b$ is brighter pixels than the intensity of center pixel $I_p$, and $P_s$ is the similar pixels to the intensity of the center pixel $I_p$

The first FAST algorithm was (FAST-12). This FAST version starts with examining the intensity of four pixels 1, 5, 9 and 13 as shown in Figure 1. If at least three out of these four pixels are brighter than $I_p + t$ or darker than $I_p - t$, then the 16 pixels are checked to find whether the 12 contiguous pixels fall in the criterion or not. If neither of these is the case, the center pixel is rejected to be a corner (Rosten and Drummond 2005). However, FAST-12 suffers from two major drawbacks (Biadgie and Sohn 2014):

- When $N < 12$, the algorithm does not work very well because the detected points will be very high.
- The speed of the algorithm depends on the order in which the 16 pixels are examined.

In order to overcome the abovementioned weakness, a machine learning technique was introduced by Rosten and Drummond (2006). This algorithm starts with detecting corners from a set of training images using the segment test criterion for a given $N$ contiguous pixels and a convenient threshold. For every pixel in the image, the circle of 16 pixels is converted into a vector and stored for further checking. Each pixel value in the vector is assigned with one of three states (darker than $I_p - t$, brighter than $I_p + t$ or similar to $I_p \pm t$). Finally, the ID3 (decision tree classifier) algorithm is employed through selecting the $x$, which represents the knowledge about whether ($I_p$) is a corner or not. This can be performed through the entropy of $K_p$ which is true if ($I_p$) is a corner and false otherwise.

Since FAST-12 has been released, various FAST versions (e.g. FAST-9, FAST-ER) have been emerged. It has been proven that FAST-9 is the most preferable version because it exhibits high performance of corner repeatability with a high test speed (Mair et al. 2010).

4. Proposed new testing procedure

As mentioned in the previous section, the original FAST algorithm performed well when the contiguous pixels which form a certain length of arc (e.g. 12, 9 pixels) are compared with the $I_p$. Then, some procedures (e.g. machine learning) are employed for speeding the test. In this research we proposed a new testing procedure based on the FAST algorithm that improves the speed of the test and provides high accuracy. As in the FAST algorithm, the proposed technique considers a circle of 16 pixels (This is a Bresenham circle of radius 3.4) surrounding the $I_p$. This circle of 16 pixels is then converted into a vector and sorted in descending/ascending order. Hence only a “single” pixel from the vector of 16 pixels is compared with the $I_p$. This single pixel is chosen based on the FAST algorithm (e.g. FAST-9, FAST-12). For instance, if FAST-9 is chosen, the single pixel will take ranking number nine in descending/ascending order.

Mathematically, the new algorithm can be written as follows:

$$S_{p_i} = \begin{cases} \text{descending} & \text{if} |I_p - I_{p_i}| \geq T \\
\text{ascending} & \text{if} |I_p - I_{p_i}| \leq -T \\
\text{otherwise} & \text{notcorner} \end{cases}$$

where $S_{p_i}$ denotes the state of pixel $i$, where pixel $i$ can be either pixel 9 or 12 corresponding to FAST-9 or FAST-12; $I_p - I_{p_i}$ is the difference between the intensity value of a pixel $i$ and the $I_p$. $T$ is the size of the threshold which can be either positive when the vector of 16 pixels are in descending order or negative in case of ascending order.
The states of the $I_P$ can be a corner in two cases. First, when the vector of 16 pixels is in descending order and the difference between pixels $i$ and the $I_P$ is equal or larger than the threshold. The second case is when the vector of 16 pixels is in ascending order and the difference between pixels $i$ and the $I_P$ is equal or less than the threshold. If neither of the above cases occurs, the $I_P$ will not be a corner.

The principles of the proposed testing procedure are shown in Figure 1. As can be seen when implementing Equation (2) and assuming that the threshold is $\pm 40$, the $I_P$ is a corner because the difference between the intensity value of the pixel number 9 in descending order and the $I_P$ is larger than the threshold (e.g. $93-50 = 43 > 40$). Hence all pixels (from 10 to 16) which are above pixel number 9 are larger than the threshold. However, pixel number 9 in ascending order is not a corner because the difference between its intensity value and the center pixel is neither equal nor less than the threshold ($94-50 = 44 < 40$).

When using descending/ascending order of the vector of 16 pixels, the arc which is formed from $N$ contiguous pixels as in the original FAST algorithm is diminished. This is because the proposed test is not restricted to the contiguous pixels but it can use $N$ discrete pixels within the vector.

\section{4.1. Evaluating the performance of the proposed test}

To evaluate the performance of the proposed algorithms for true extraction of crowd features, completeness and correctness (Heipke et al. 1997; Rottensteiner et al. 2007) are commonly used. Both of which are written as follows:

$$\text{completeness} = \frac{TP}{TP + FN} \quad (3)$$

$$\text{correctness} = \frac{TP}{TP + FP} \quad (4)$$

where $TP$ denotes the number of true positives, i.e. the number of entities found to be available in both reference and experiment datasets; $FN$ is the number of false negatives, i.e. the number of entities in the reference dataset that were not detected automatically, and $FP$ is the number of false positives, i.e. the number of entities that were detected, but do not correspond to an entity in the reference dataset.

In this research, the original FAST-9 and FAST-12 have been used as reference algorithms to compare the performance of the proposed test with both of them.

\section{4.2. Filtering the crowd features}

Following the feature extraction procedures which are presented in the previous section, a new step is necessary to discern between the crowd features and other feature classes. This means exclusion of other feature classes which do not belong to the crowd features. In this research, we assume that the crowd features are clustered a long open areas (e.g. streets). Thus, feature points which occur in surrounding environments do not belong to the crowd and they should be eliminated. In order to avoid arbitrary exclusion of non-crowd features, a procedure is introduced herein. This procedure begins with plotting the extracted crowd features along X/Y axes. A center point is then determined based on a K-means clustering algorithm. From the center point, a circle is created based on a predefined radius and a certain number of sectors. Consequently, each sector forms a region that includes a certain number of feature points.

When defining the minimum number of feature points inside the region, some feature points will be eliminated because they are less than the minimum required points. Although this filtering procedure may not suit the sparse crowd features, it is still efficient for dense crowd regions. For instance, when a region size of 1 m$^2$ includes one person (one person/1 m$^2$), this region is eliminated because it includes sparse crowd. On the other hand, when the same region (1 m$^2$) includes 3–4 people, a dense region cannot be eliminated because it is crowded region. This procedure is shown in Figure 2. It can clearly be seen that some regions are very dense while other regions contain very few feature points. If for example, the number of feature points in a region does not meet the minimum requirement points, those feature points will be excluded and vice versa.

\section{5. Mapping the levels of crowd density}

After exclusion of the unnecessary feature points, the remaining coordinates of feature points $(x_i, y_i) \ i \in [1, 2, \ldots, K]$ have been used as inputs in 2-D Gaussian Kernel density estimation (KDE). This step is to estimate the probability density function and hence map the levels of the crowd density. The estimated probability density function can be written as follows (Butenuth et al. 2011; Fradi and Dugelay 2013):

$$p(x, y) = \frac{1}{\sqrt{2\pi} \sigma} \sum_{i=1}^{K} \exp \left( - \frac{(x - x_i)^2 + (y - y_i)^2}{2\sigma^2} \right) \quad (5)$$

where $\sigma$ is the bandwidth of the Gaussian kernel.
6. Test and results

6.1. Test

The UAV images that show the distribution of the crowd have been extracted from online video and used in this paper after pre-processing procedures including trimming and resizing. Crowd images were then treated as a function of camera position and orientation. Thus, vertical and horizontal (long and close range) images were used. Figures 3–5 illustrate the types of images used in this study. The specifications of these images are also shown in Table 1.

Afterwards, crowd features have been extracted using a new testing algorithm based on the FAST method. Both FAST-9 and FAST-12, based on the new testing procedure, were employed to detect corner points which mainly represent crowd features. Then, filtering procedures were used to eliminate the feature points which do not belong to crowd features. Ultimately, the levels of crowd density were mapped using 2-D Gaussian KDE.

6.2. Results

6.2.1. Extraction of crowd features using new testing procedure

Extracted crowd features have been presented in Figure 6–8 using the new testing procedure in both FAST-9 and FAST-12. As can be seen, the proposed testing algorithm was successful in discerning the crowd features from other feature classes. As a normal situation, when using FAST-9 (red points) with a threshold of ± 40, the number of detected corners is higher than those detected by FAST-12. It is obvious that FAST-12 (green points) detect very strong corner points.

In comparison between vertical and horizontal UAV images, one can note that when using vertical images, the results which represent actual crowd
features were relatively more accurate than those presented in both horizontal images. This is because the camera was focusing on an area covered by the crowd. In other image cases however, the surrounding environment (trees, streets and buildings) has taken a huge part of the image. As can be seen in Figures 7 and 8 which show detected corner points in horizontal images, some of the detected corners were flagged as crowd features while they are actually not. This is because the intensity values of those pixels (features from the surrounding environment e.g. building, streets and trees) have similar intensity values to those of crowd features.

Figures 9 and 10 illustrate a comparison between the proposed algorithms and both of the original FAST-9 and FAST-12 with different thresholds values respectively. In general, the number of detected crowd features when using the proposed algorithm is higher than those detected by the original FAST. This is also clear in Table 2 which shows detailed comparison between the number of detected crowd features as a function of threshold values in FAST-9, FAST-12 methods and the proposed algorithm.

The performance of the proposed procedure is shown in Figures 11 and 12. According to Figure 11, completeness and correctness were shown as a function of threshold. The notable situation is an oppositional behavior of completeness and correctness when they response to the size of threshold. A

| Image orientation   | Camera angle | Image size in pixels |
|---------------------|--------------|----------------------|
| Vertical            | 90           | 691 × 1339 × 3 unit8 |
| Horizontal (close range) | 45           | 683 × 471 × 3 unit8 |
| Horizontal (long range) | 45           | 689 × 1366 × 3 unit8 |

Figure 6. Crowd feature extraction from horizontal (close range) UAV image using the new testing procedure (image size in pixels).

Figure 7. Crowd feature extraction from vertical UAV image using the new testing procedure (image size in pixels).

Figure 8. Crowd feature extraction from horizontal UAV image (long range) using the new testing procedure (image size in pixels).

Figure 9. Number of detected crowd features using both the original FAST-9 and the proposed algorithm.

Figure 10. Number of detected crowd features using both the original FAST-9 and the proposed algorithm.

Table 1. UAV images specifications.

| Image orientation   | Camera angle | Image size in pixels |
|---------------------|--------------|----------------------|
| Vertical            | 90           | 691 × 1339 × 3 unit8 |
| Horizontal (close range) | 45           | 683 × 471 × 3 unit8 |
| Horizontal (long range) | 45           | 689 × 1366 × 3 unit8 |
low value of threshold has significant negative influence on the percentage of correctness whereas completeness is positively influenced. An instance is shown in Figure 11 that illustrates a comparison between the original FAST-9 and the proposed test. It is obvious that when the value of threshold is ±10 the completeness is around 100% while correctness is approximately 1%. On the contrary, when the value of threshold is ±100 the percentage of completeness is around 1% comparing with 100% for correctness. Furthermore, the highest percentage of correctness and completeness (76%) has been achieved when the value of threshold is ±40. Similar situation can be seen when comparing between the original FAST-12 and the proposed test (Figure 12). In this case however, both completeness and correctness are in their highest values (around 80% when the value of threshold is between ±35 and ±40). Thus, it is necessary to select a proper value of threshold that accurately detects the crowd features.

6.2.2. Filtering the crowd features

It is highly important to select the corner points which actually represent the crowd. Some feature points which deviate markedly from the dense crowd regions need to be eliminated. Figure 13(a) show the distribution of feature points before and after filtering, respectively. As one might note, the remaining feature points in Figure 13(b) are mostly related to the crowd region. The percentage of the corrected crowd after filtering was around 87%. Thus, 13% of point features were eliminated. In some cases however, very few feature points which represent people were also eliminated. This is because the location of those features is far away from the region of crowd features.

It is clear that the performance of the filtering method introduced in this paper corresponds to the number of minimum points in a sector, the number of sectors in a circle and the length of the circle radius. Therefore, a comparison between the original FAST-12 algorithm and the new testing procedure after implementing the filtering method was conducted. This can be seen in Table 3 which shows the accuracy assessment in terms of completeness and correctness as a function of the number of minimum points in a sector, the number of sectors in a circle and the length of radius. Furthermore, the
threshold value was 30 for the original FAST-12 and ± 40 for the new testing method. As can be seen, the behavior of completeness and correctness is similar to those presented in the previous section. In addition, a slight change in the percentage of completeness and correctness can be seen when changing the settings of the filtering method. Under all circumstances, the values of completeness range from 55 to 70% whereas the range of correctness values was 91 to 94%.

6.2.3. Mapping the levels of crowd density

2-D KDE before and after filtering the crowd features is illustrated in Figure 14 (a) and (b) respectively, contour lines were drawn around the dense regions and highlight the levels of crowd density using gradual color (blue to red) where blue and red show the lowest and highest values respectively. Generally speaking, the levels of crowd density were low to medium in almost all crowd regions. The cause of many sparse crowd regions is attributed to the gaps between people that yield a number of unconnected feature points.

6.2.4. Performance of the proposed test under different light circumstances

Under different light circumstances, the performance of the proposed algorithms was examined. Figure 15 (a), (b) and (c) demonstrate the influences of the light conditions such as normal, bright and dark light on the number of detected crowd features respectively. Despite of using the same image and same threshold value (60) under all these light conditions, the number of detected crowd features was entirely different. As can be seen in Table 4, the number of detected crowd features under normal light conditions, was 3893 and 777 in both FAST-9 and FAST-12 respectively. When the image is dark the number of detected crowd features

---

Table 3. Completeness and correctness for FAST-9 and FAST-12 using the new testing procedure under different filtering cases (number of minimum points in each sector, number of sectors in each circle and length of radius).

| Circle Radius | Number of sectors | Number of minimum points in sector | Threshold of reference algorithm | Threshold of proposed algorithm | Completeness % | Correctness % |
|---------------|-------------------|------------------------------------|---------------------------------|--------------------------------|---------------|---------------|
| 100           | 4                 | 30 points                          | FAST-12                         | FAST-12                         | 0.69          | 0.92          |
|               |                   | 60 points                          |                                 |                                | 0.66          | 0.93          |
|               |                   | 90 points                          |                                 |                                | 0.62          | 0.94          |
|               | 6                 | 30 points                          | 30                              | 40                             | 0.67          | 0.93          |
|               |                   | 60 points                          |                                 |                                | 0.65          | 0.93          |
|               |                   | 90 points                          |                                 |                                | 0.62          | 0.94          |
|               | 8                 | 30 points                          | 30                              | 40                             | 0.68          | 0.92          |
|               |                   | 60 points                          |                                 |                                | 0.61          | 0.94          |
|               |                   | 90 points                          |                                 |                                | 0.55          | 0.94          |
| 200           | 4                 | 30 points                          | 30                              | 40                             | 0.70          | 0.91          |
|               |                   | 60 points                          |                                 |                                | 0.69          | 0.91          |
|               |                   | 90 points                          |                                 |                                | 0.69          | 0.91          |
|               | 6                 | 30 points                          | 30                              | 40                             | 0.69          | 0.92          |
|               |                   | 60 points                          |                                 |                                | 0.68          | 0.92          |
|               |                   | 90 points                          |                                 |                                | 0.68          | 0.92          |
|               | 8                 | 30 points                          | 30                              | 40                             | 0.70          | 0.91          |
|               |                   | 60 points                          |                                 |                                | 0.68          | 0.92          |
|               |                   | 90 points                          |                                 |                                | 0.68          | 0.92          |
decreases to around 1566 and 313, whereas the bright image yields 14,250 and 3966 feature points in both FAST-9 and FAST-12 respectively. Therefore, light intensity should be considered in terms of camera type and platform (static or dynamic), surrounding environment, and the time the images are taken when detecting crowd features from images. This will enhance selecting an optimum size of threshold that reflects accurate results.

7. Conclusions

Previous research on crowd analysis has been directed toward estimating the levels of crowd density. As a result, developing and improving the existing approaches such as background removal (Davies, Yin, and Velastin 1995) image processing (Marana et al. 1998) and corner detection approaches (Butenuth et al. 2011) has taken a considerable attention to introduce an optimum method for crowd analysis. Among these methods is the FAST (Rosten and Drummond 2005) which has been used for detecting crowd features from land-based and aerial-based images.

In this research, a new testing algorithm based on the FAST method has been introduced and implemented to detect crowd features from different UAV images. Vertical and horizontal UAV images were employed to detect crowd features and for mapping the levels of crowd density. The performance of the new testing algorithms was compared with the original FAST algorithms to assess the accuracy in terms of completeness and correctness. Crowd features were then filtered through eliminating the feature points which deviate from the clustered crowd regions. The final step was mapping the levels of crowd density using two-dimensional kernel density estimation.

In complex environments where many feature classes appear in an image, detecting crowd features is a difficult task. The results show that the proposed testing algorithms were able to detect crowd features among other feature classes. This has been seen in all UAV image cases. Furthermore, the size of the threshold has a great impact on completeness and correctness as such a large threshold yields low values of completeness and high values of correctness. Therefore, this research emphasis the selection of optimum size of threshold that detects true feature points without false corners. A large threshold value yields very few strong corners while a small value detects many false corner points. Thus, finding a proper threshold is a trade-off between the number of non-detected true corner points and many detected false corners (Mair et al. 2010; Vino and Sappa 2013). According to these research results, the value of the threshold influences the accuracy of the detected crowd features as when the value of threshold was small (e.g. 10), 100 % of completeness was achieved. This is because a huge number of crowd features were detected with the new testing algorithms and the probability to intersect with the crowd features in the reference dataset is very high. On the contrary, correctness becomes very low due to the fact that the detected crowd features do not correspond to the crowd features in the reference dataset.
On the other hand, the situation becomes entirely different when using a large threshold. As an instance, the completeness has the lowest value while correctness becomes very high. It has been shown (Figures 11 and 12) that a threshold of around (± 40) is a suitable value to achieve accurate results.

However, the value of threshold should coincide with light intensities. This is because the accuracy will be degraded if one threshold value is used under different light conditions. This has been shown in Figure 15 and Table 4. As a result, light intensity in terms of the surrounding environment, camera type

Figure 15. Performance of the proposed algorithms under (a) normal light, (b) dark, and (c) bright conditions (image size in pixels).
Table 4. Performance of FAST method for detecting crowd features under different cases of light conditions.

| FAST version | Normal Image | Brighter Image | Darker Image |
|--------------|--------------|---------------|--------------|
| Proposed FAST-9 | 3893 | 14,250 | 1566 |
| Proposed FAST-12 | 777 | 3966 | 313 |

* The threshold used in all cases was ± 60

and platform, and the time of the images are taken need to be considered when extracting crowd features from images.

In this paper, a filtering procedure has been introduced to select the feature points which actually represent crowd features. As presented in this research, the filtering procedure was able to eliminate some feature points particularly those that deviate markedly from the overall distribution of the feature points. This is because the number of those feature points was less than the predefined minimum numbers of points in the circle sector. The accuracy assessment has been slightly improved when implementing the filtering procedures (Table 2).

Future research will focus on detecting crowd features using geo-referenced images to generate real maps for crowd density. Furthermore, a new testing procedure using the FAST algorithm will be further investigated through using an adaptive threshold that provides high level of accuracy.

Acknowledgments

My sincere thanks go to Associate Professor Jinling Wang from University of New South Wales, Sydney-Australia for providing academic advices and support to achieve this research. I would like also express my deepest appreciation to assistant Professors Khalid Hazaymeh and Mohammad Zeitoun from Yarmouk University for their support.

Notes on contributor

Ali Almagbile is an assistant professor at the Department of Geography at Yarmouk University, Jordan. He received his PhD in surveying and spatial information systems from the University of New South Wales (UNSW), Sydney-Australia. His research interests cover multi-sensor integration systems (GNSS/INS/Vision sensors), quality control (reliability theory), photogrammetry and UAVs surveying and mapping.

References

Biadgie, Y., and K. A. Sohn. 2014. "Feature Detector Using Adaptive Accelerated Segment Test." Paper Presented at 5th International Conference on Information Science and Applications, Seoul, May 6–9.

Burkert, F., and M. Butenuth. 2011. "Event Detection Based on a Pedestrian Interaction Graph Using Hidden Markov Models." Paper Presented at 2011 ISPRS conference on Photogrammetric image analysis, Munich, Germany, October 5–7.

Butenuth, M., F. Burkert, F. Schmidt, S. Hinz, D. Hartmann, A. Kneidl, A. Borrmann, and B. Sirmacek. 2011. "Integrating Pedestrian Simulation, Tracking and Event Detection for Crowd Analysis." Paper Presented at IEEE International Conference on Computer Vision Workshops, Barcelona, Spain, November 6–13.

Conte, D., P. Foggia, G. Percannella, F. Tufano, and M. Vento. 2010. "A Method for Counting People in Crowded Scenes." Paper Presented at 7th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), Boston, Massachusetts, U.S, August 29–September 1.

Davies, A., J. Yin, and S. Velastin. 1995. "Crowd Monitoring Using Image Processing." Electronics & Communication Engineering Journal 7 (1): 37–47. doi:10.1049/ecej:19950106.

Fradi, H., and J. L. Dugelay. 2013. "Crowd Density Map Estimation Based on Feature Tracks." Paper Presented at MMSP 15th International Workshop on Multimedia Signal Processing, Pula, Italy, September 30–October 2.

Harris, C., and M. Stephens. 1988. Acombined Corner and Edge Detector. Paper Presented at 4th Alvey Vision Conference, Manchester, UK, August 31–September 2. doi:10.3168/jsd.80022-0302(88)79586-7.

Heipke, C., H. Mayer, C. Wiedemann, and O. Jamet. 1997. "Evaluation of Automatic Road Extraction." International Archives of Photogrammetry and Remote Sensing, 32 (3-2W3): 47–56.

Jiang, M., J. Huang, X. Wang, and J. Tang. 2014. "An Approach for Crowd Density and Crowd Size Estimation." Journal of Software 9 (3): 757–762. doi:10.4304/jsw.9.3.757-762.

Lambers, K., H. Eisenbeiss, M. Saurerbier, D. Kupferschmidt, T. Gaisecker, S. Sotoodeh, and T. Hanusch. 2007. "Combining Photogrammetry and Laser Scanning for the Recording and Modelling of the Late Intermediate Period Site of Pinchango Alto, Palpa, Peru." Journal of Archaeological Science 34 (10): 1702–1712. doi:10.1016/j.jas.2006.12.008.

Mair, E., G. Hager, D. Burschka, M. Suppa, and G. Hirzinger. 2010. "Adaptive and Generic Corner Detection Based on the Accelerated Segment Test." Paper Presented at 11th European Conference on Computer Vision: Part II, Heraklio, Crete, Greece, September 5–11.

Marana, A., S. Velastin, L. Costa, and R. Lotufo. 1998. "Automatic Estimation of Crowd Density Using Texture." Safety Science 28 (3): 165–175. doi:10.1016/S0925-7535(97)00081-7.

Nagai, M., T. Chen, R. Shibasaki, H. Kumugai, and A. Ahmed. 2009. "UAV-borne 3-D Mapping System by Multi-Sensory Integration." IEEE Transactions on Geoscience and Remote Sensing 47 (3): 701–708. doi:10.1109/TGRS.2008.2010314.

Patterson, M., and A. Brescia. 2008. "Integrated Sensor Systems for UAS." Paper Presented at 23rd Bristol International Unmanned Air Vehicle Systems (UAVS) Conference, Bristol, United Kingdom, April 7–9.

Rosten, E., and T. Drummond. 2005. "Fusing Points and Lines for High Performance Tracking." Paper Presented at 10th IEEE International Conference on Computer Vision, Beijing, China, October 17–21.

Rosten, E., and T. Drummond. 2006. "Machine Learning for High Speed Corner Detection." Paper Presented at 9th Europan Conference on Computer Vision, Graz, Austria, May 7–13.
Rosten, E., R. Porter, and T. Drummond. 2010. “FASTER and Better: A Machine Learning Approach to Corner Detection.” IEEE Transactions Pattern Analysis and Machine Intelligence 32: 105–119. doi:10.1109/TPAMI.2008.275.

Rottensteiner, F., J. Trinder, S. Clode, and K. Kubik. 2007. “Building Detection by Fusion of Airborne Laser Scanner Data and Multi-Spectral Images: Performance Evaluation and Sensitivity Analysis.” ISPRS Journal of Photogrammetry and Remote Sensing 62: 135–149. doi:10.1016/j.isprsjprs.2007.03.001.

Sirmacek, B., and P. Reinartz. 2011. “Automatic Crowd Analysis from Very High Resolution Satellite Images.” Paper Presented at Photogrammetric Image Analysis Conference (PIA11) Munich, Germany, October 5–7.

Velastin, S., J. Yin, A. Davies, M. Vicencio–Silva, R. Allsop, and A. Penn. 1994. “Automated Measurement of Crowd Density and Motion Using Image Processing.” Paper Presented at 7th International Conference Road Traffic Monitoring and Control, London, UK, April 26–28. doi:10.3168/jds.S0022-0302(94)77044-2

Vino, G., and A. D. Sappa. 2013. “Revisiting Harris Corner Detector Algorithm: A Gradual Thresholding Approach.” Paper Presented at International Conference on Image Analysis and Recognition, Povoa de Varzim, Portugal, June 26–28.

Wang, B., H. Bao, S. Yang, and H. Lou. 2013. “Crowd Density Estimation Based on Texture Feature Extraction.” Journal of Multimedia 8 (4): 331–337. doi:10.4304/jmm.8.4.331-337.

Xu, C., H. Bao, Z. Lulu, and N. He. 2014. “Crowd Density Estimation Based on Improved Harris & OPTICS Algorithm.” Journal of Computers 9 (5): 1209–1217. doi:10.4304/jcp.9.5.1209-1217.

Yang, D., H. Gonzalez-Banos, and L. Guibas. 2003. “Counting People in Crowds with a Real-Time Network of Simple Image Sensors.” Paper Presented at 9th IEEE International Conference on Computer Vision, Nice, France, October 13–16.

Yin, J., S. Velastin, and A. Davies. 1995. “Image Processing Techniques for Crowd Density Estimation Using a Reference Image.” Paper presented at 2nd Asia–Pacific Conference on Computer Vision, Singapore, December 5–8.

Zhan, B., D. N. Monekosso, P. Remagnino, S. A. Velastin, and L. Xu. 2008. “Crowd Analysis: A Survey.” Machine Vision and Applications 19 (5–6): 345–357. doi:10.1007/s00138-008-0132-4.