KALMAN FILTER BASED FEATURE ANALYSIS FOR TRACKING PEOPLE FROM AIRBORNE IMAGES

Beril Sirmacek, Peter Reinartz

German Aerospace Center (DLR), Remote Sensing Technology Institute
PO Box 1116, 82230, Wessling, Germany
(Beril.Sirmacek)@dlr.de

Commission III/5

KEY WORDS: Image sequences, Airborne images, People tracking, Tracking multiple objects, Feature extraction, Feature tracking, Kalman filtering, Object detection

ABSTRACT:

Recently, analysis of man events in real-time using computer vision techniques became a very important research field. Especially, understanding motion of people can be helpful to prevent unpleasant conditions. Understanding behavioral dynamics of people can also help to estimate future states of underground passages, shopping center like public entrances, or streets. In order to bring an automated solution to this problem, we propose a novel approach using airborne image sequences. Although airborne image resolutions are not enough to see each person in detail, we can still notice a change of color components in the place where a person exists. Therefore, we propose a color feature detection based probabilistic framework in order to detect people automatically. Extracted local features behave as observations of the probability density function (pdf) of the people locations to be estimated. Using an adaptive kernel density estimation method, we estimate the corresponding pdf. First, we use estimated pdf to detect boundaries of dense crowds. After that, using background information of dense crowds and previously extracted local features, we detect other people in non-crowd regions automatically for each image in the sequence. We benefit from Kalman filtering to track motion of detected people. To test our algorithm, we use a stadium entrance image data set taken from airborne camera system. Our experimental results indicate possible usage of the algorithm in real-life man events. We believe that the proposed approach can also provide crucial information to police departments and crisis management teams to achieve more detailed observations of people in large open area events to prevent possible accidents or unpleasant conditions.

1 INTRODUCTION

Recently automatic detection of people and understanding their behaviors from images became a very important research field, since it can provide crucial information especially for police departments and crisis management teams. Tracking people, understanding their moving directions and speeds can be used for detecting abnormal situations. Besides, it can also help to estimate locations where a crowd can congregate which gives idea about future states of underground passages, shopping center like public entrances, or streets which can also affect the traffic.

Due to the importance of the topic, many researchers tried to monitor behaviors of people using street, or indoor cameras which are also known as close-range cameras. However, most of the previous studies aimed to detect boundaries of large groups, and to extract information about them. The early studies in this field were developed from closed-circuit television images (Davies et al., 1995), (Regazzoni and Tesei, 1994), (Regazzoni and Tesei, 1996). Unfortunately, these cameras can only monitor a few square meters in indoor regions, and it is not possible to adapt those algorithms to street or airborne cameras since the human face and body contours will not appear as clearly as in close-range indoor camera images due to the resolution and scale differences. In order to be able to monitor bigger events researchers tried to develop algorithms which can work on outdoor camera images or video streams. Arandjelovic (Arandjelovic, Sep. 2008) developed a local interest point extraction based crowd detection method to classify single terrestrial images as crowd and non-crowd regions. They observed that dense crowds produce a high number of interest points. Therefore, they used density of SIFT features for classification. After generating crowd and non-crowd training sets, they used SVM based classification to detect crowds. They obtained scale invariant and good results in terrestrial images. Unfortunately, these images do not enable monitoring large events, and different crowd samples should be detected before hand to train the classifier. Ge and Collins (Ge and Collins, 2009) proposed a Bayesian marked point process to detect and count people in single images. They used football match images, and also street camera images for testing their algorithm. It requires clear detection of body boundaries, which is not possible in airborne images. In another study, Ge and Collins (Ge and Collins, 2010) used multiple close-range images which are taken at the same time from different viewing angles. They used three-dimensional heights of the objects to detect people on streets. Unfortunately, it is not always possible to obtain these multi-view close-range images for the street where an event occurs. Chao et al. (Lin et al., Nov. 2001) wanted to obtain quantitative measures about crowds using single images. They used Haar wavelet transform to detect head-like contours, then using SVM they classified detected contours as head or non-head regions. They provided quantitative measures about number of people in crowd and sizes of crowd. Although results are promising, this method requires clear detection of human head contours and a training of the classifier. Unfortunately, street cameras also have a limited coverage area to monitor large outdoor events. In addition to that, in most of the cases, it is not possible to obtain close-range street images or video streams in the place where an event occurs. Therefore, in order to behaviors of large groups of people in very big outdoor events, the best way is to use airborne images which began to give more information to researchers with the development of sensor technology. Since most of the previous approaches in this field needed clear detection of face or body features, curves, or boundaries to detect people and crowd boundaries which is not possible in airborne images, new approaches are needed to extract information from these images. In a previous study Hinz et al. (Hinz, 2009) registered airborne image sequences to estimate density and motion of people in crowded regions. For this purpose, first a training background segment is selected manually to classify image as foreground and background pixels. They used the ratio of background pixels and foreground pixels in a neighborhood to plot density map. Observing change of the density map in the sequence, they estimated motion of people. Unfortunately, their approach did not provide quantitative measures about crowds. In a following study (Burkert et al., Sep. 2010), they used previous approach to detect individuals. Positions of detected people are linked with graphs. They used these graphs for understanding behaviors of people.

In order to bring an automated solution to the problem, herein we propose a novel automatic framework to track people and to understand their behaviors from airborne images. For this purpose, first we introduce our automatic crowd and people detection approach which is based on local features extracted from chroma bands of the input image. After detecting dense crowd regions and people in sparse groups, we apply tracking using Kalman filter. Our experiments on registered color airborne image sequences indicate possible usage of the proposed framework in real-life applications. We believe that proposed dense crowd detection, people detection, and people tracking approaches can provide crucial information to police departments and crisis management teams to prevent possible accidents or unpleasant conditions.

2 DETECTING PEOPLE FROM AIRBORNE IMAGES

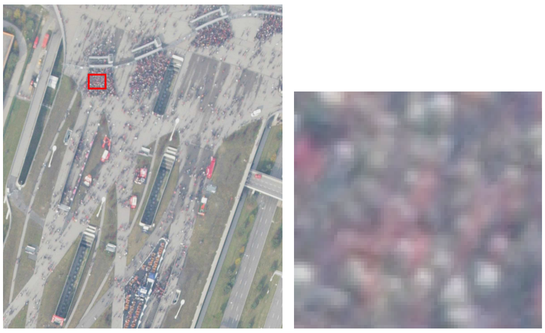
For each airborne image in the input sequence, before tracking process, we apply dense crowd detection and people detection approach. Next, we introduce steps of the approach in detail.

2.1 Local Feature Extraction

In order to illustrate the algorithm steps, we pick $Stadium_1$ image from our $Stadium_{1-43}$ test image sequence. In Fig.1.(a), we represent $Stadium_1$ test image, and in Fig.1.(b), we represent a subpart of the original image in order to give information about real resolution of the image. As can be seen here, airborne image resolutions do not enable to see each single person with sharp details. However, we can still notice a change of color components in the place where a person exists. Therefore, our dense crowd and people detection method depends on local features extracted from chroma bands of the input test image.

For local feature extraction, we use features from accelerated segment test (FAST). FAST feature extraction method is especially developed for corner detection purposes by Rosten et al. (Rosten et al., Nov. 2010), however it also gives high responses on small regions which are significantly different than surrounding pixels. The method depends on wedge-model-style corner detection and machine learning techniques. For each feature candidate pixel, its 16 neighbors are checked. If there exist nine contiguous pixels passing a set of pixels, the candidate pixel is labeled as a feature location. In FAST method, these tests are done using machine learning techniques to speed up the operation.

For FAST feature extraction from invariant color bands of the image, we first start with converting our RGB test image into CIELab color space. In many computer applications, the CIELab color space is used since it mimics the human visual system. CIELab color space bands are able to enhance different colors best and minimize color variances (Fairchild, 1998). After transforming the RGB color image into CIELab color space, again we obtain three bands as L, a, and b (Paschos, 2001). Here, L band corresponds to intensity of the image pixels. a, b bands contain chroma features of the image. These two bands give information about the color information independent of illumination. For illumination invariance, in this study we use only a and b chroma bands of image for local feature extraction. To detect small regions which have significantly different color values than their surroundings, we extract FAST features from a and b chroma bands of the image. For detailed explanation of FAST feature extraction method please see (Rosten et al., Nov. 2010).



(a) (b)

Figure 1: (a) $Stadium_1$ test image from our airborne image sequence including both crowded and sparse people groups, (b) Closer view of a crowded region in $Image_1$.

We assume (x_a, y_a) $a \in [1, 2, ..., K_a]$ and (x_b, y_b) $b \in [1, 2, ..., K_b]$ as FAST local features which are extracted from a and b chroma bands of the input image respectively. Here, K_a and K_b indicates the maximum number of features extracted from each chroma band. As local feature, in our study we use (x_i, y_i) $i \in [1, 2, ..., K_i]$ features which holds features coming from two chroma bands. However, if two features from different bands are extracted at the same coordinates, it is held only for one time in (x_i, y_i) $i \in [1, 2, ..., K_i]$ array. Therefore, we expect K_i number to be less than or equal to $K_a + K_b$.

We represent locations of detected local features for $Stadium_1$ test image in Fig. 2.(a). Extracted FAST features behave as observations of the probability density function (pdf) of the people to be estimated. In the next step, we introduce an adaptive kernel density estimation method, to estimate corresponding pdf which will help us to detect dense people groups and people in dense groups.

2.2 Detecting Dense Crowds Based on Probability Theory

Since we have no pre-information about the street, building, green area boundaries and crowd locations in the image, we formulate the crowd detection method using a probabilistic framework. Assume that (x_i, y_i) is the ith FAST feature where $i \in [1, 2, ..., K_i]$. Each FAST feature indicates a local color change which might be a human to be detected. Therefore, we assume each FAST feature as an observation of a crowd pdf. For crowded regions, we assume that more local features should come together. Therefore knowing the pdf will lead to detection of crowds. For pdf estimation, we benefit from a kernel based density estimation method as Sirmacek and Unsalan represented for local feature based building detection (Sirmacek and Unsalan, 2010).

Silverman (Silverman, 1986) defined the kernel density estimator for a discrete and bivariate pdf as follows. The bivariate kernel function $[N(x, y)]$ should satisfy the conditions given below;

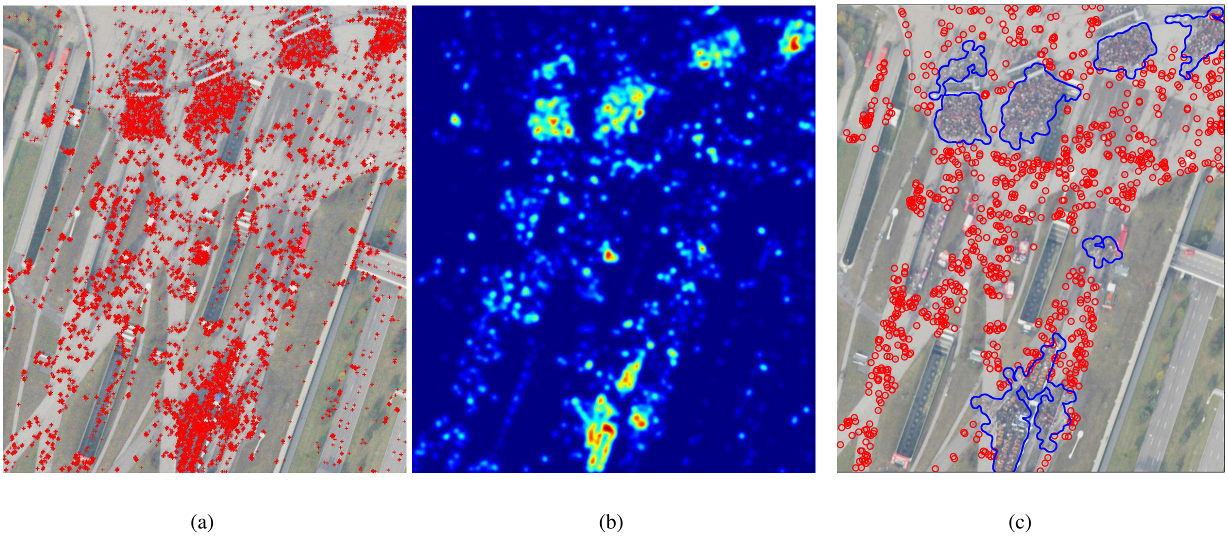


Figure 2: (a) Detected FAST feature locations on $Stadium_1$ test image are represented with red crosses, (b) Estimated probability density function (color coded) for $Stadium_1$ image generated using FAST feature locations as observations, (c) Automatically detected dense crowd boundaries and detected people in sparse groups for $Stadium_1$ image.

$$\sum_x \sum_y N(x,y) = 1 \quad (1)$$

$$N(x,y) \geq 0, \forall(x,y) \quad (2)$$

The pdf estimator with kernel $N(x,y)$ is defined by,

$$p(x,y) = \frac{1}{nh}\sum_{i=1}^{n} N(\frac{x-x_i}{h}, \frac{y-y_i}{h}) \quad (3)$$

where h is the width of window which is also called smoothing parameter. In this equation, (x_i, y_i) for $i = 1, 2, ..., n$ are observations from pdf that we want to estimate. We take $N(x,y)$ as a Gaussian symmetric pdf, which is used in most density estimation applications. Then, the estimated pdf is formed as below;

$$p(x,y) = \frac{1}{R}\sum_{i=1}^{K_i} \frac{1}{\sqrt{2\pi\sigma}} exp(-\frac{(x-x_i)^2 + (y-y_i)^2}{2\sigma}) \quad (4)$$

where σ is the bandwidth of Gaussian kernel (also called smoothing parameter), and R is the normalizing constant to normalize $p_n(x,y)$ values between $[0,1]$.

In kernel based density estimation the main problem is how to choose the bandwidth of Gaussian kernel for a given test image, since the estimated pdf directly depends on this value. For instance, if the resolution of the camera increases or if the altitude of the plane decreases, pixel distance between two persons will increase. That means, Gaussian kernels with larger bandwidths will make these two persons connected and will lead to detect them as a group. Otherwise, there will be many separate peaks on pdf, but we will not be able to find large hills which indicate crowds. As a result, using a Gaussian kernel with fixed bandwidth will lead to poor estimates. Therefore, bandwidth of Gaussian kernel should be adapted for any given input image.

In probability theory, there are several methods to estimate the bandwidth of kernel functions for given observations. One well-known approach is using statistical classification. This method is based on computing the pdf using different bandwidth parameters and then comparing them. Unfortunately, in our field such a framework can be very time consuming for large input images. The other well-known approach is called balloon estimators. This method checks k-nearest neighborhoods of each observation point to understand the density in that area. If the density is high, bandwidth is reduced proportional to the detected density measure. This method is generally used for variable kernel density estimation, where a different kernel bandwidth is used for each observation point. However, in our study we need to compute one fixed kernel bandwidth to use at all observation points. To this end, we follow an approach which is slightly different from balloon estimators. First, we pick $K_i/2$ number of random observations (FAST feature locations) to reduce the computation time. For each observation location, we compute the distance to the nearest neighbor observation point. Then, the mean of all distances give us a number l (calculated 105.6 for $Stadium_1$). We assume that variance of Gaussian kernel (σ^2) should be equal or greater than l. In order to guarantee to intersect kernels of two close observations, we assume variance of Gaussian kernel as $5l$ in our study. Consequently, bandwidth of Gaussian kernel is estimated as $\sigma = \sqrt{5l}$. For a given sequence, that value is computed only one time over one image. Then, the same σ value is used for all observations which are extracted from images of the same sequence. The introduced automatic kernel bandwidth estimation method, makes the algorithm robust to scale and resolution changes.

In Fig. 2.(b), we represent obtained pdf for $Stadium_1$ test image. Represented pdf function is color coded, which means yellow-red regions show high probability values, and dark blue regions show low probability values. As can be seen in this figure, crowded areas have very high probability values, and they are highlighted in estimated pdf. We use Otsu's automatic thresholding method on this pdf to detect regions having high probability values (Otsu, 2009). After thresholding our pdf function, in obtained binary image we eliminate regions with an area smaller than 1000 pixels since they cannot indicate large human crowds. The resulting binary image $B_c(x,y)$ holds dense crowd regions. For $Stadium_1$

image, boundaries of detected crowd regions are represented on original input image with blue borders in Fig. 2.(c). After detecting very dense people groups, in the next step we focus on detecting other people in sparse groups.

After detecting dense crowds automatically, we also extract quantitative measures from detected crowds for more detailed analysis. Since they indicate local color changes, we assume that detected features can give information about number of people in crowded areas. Unfortunately, number of features in a crowd region do not give the number of people directly. In most cases, shadows of people or small gaps between people also generate a feature. Besides, two neighbor features might come from two different chroma bands for the same person. In order to decrease counting errors coming from these features, we follow a different strategy to estimate the number of people in detected crowds. We use a binary mask $B_f(x, y)$ where feature locations have value 1. Then, we dilate $B_f(x, y)$ using a disk shape structuring element with a radius of 2 to connect close feature locations. Finally, we apply connected component analysis to mask, and we assume the total number connected components which are laying in a crowd area as the number of people (N). In this process, slight change of radius of structuring element does not make a significant change in estimated people number N. However, an appreciable increase in radius can connect features coming from different persons, and that decreases N which leads to poor number of people estimates.

If the resolution of the input image is known, using estimated number of people in crowd, the density of people (d) can also be calculated. Lets assume, $B_c^j(x, y)$ is the jth connected component in $B_c(x, y)$ crowd mask. We calculate crowd density for jth crowd as $d^j = N/(\sum_X \sum_Y B_c^j(x, y) \times a)$, where X and Y are the numbers of pixels in the image in horizontal and vertical directions respectively, and a is the area of one pixel as m^2.

2.3 Detecting People in Sparse Groups

Besides detecting dense crowd regions and extracting quantitative measures on them, detecting other people in non-crowd regions is also crucial. Since detecting people in non-crowd regions can help to develop people tracking or behavior understanding systems.

In order to detect people in non-crowd regions, we apply connected component analysis (Sonka et al., 2007) to $B_f(x, y)$ matrix, and pick mass centers of the connected components (x_p, y_p) $p \in [1, 2, ..., K_p]$ which satisfy $B_c(x_p, y_p) = 0$ as locations of individual people in sparse groups. Unfortunately, each (x_p, y_p) $p \in [1, 2, ..., K_p]$ location satisfying this rule does not indicate a person appearance directly. Since the location might be coming from irrelevant local features coming of another object like a tree or chimney of a rooftop. In order to decide that, if a (x_p, y_p) position is indicating a person appearance or not, we apply a background comparison test. At this step, in order to represent a person, background color of a connected component which is centered in (x_p, y_p) position should be very similar to the background color of detected dense crowds.

In order to do background similarity test, first we pick all border pixels of the binary objects (crowd regions) in $B_c(x, y)$ binary crowd mask. We assume L_c, a_c, and b_c as mean of L, a, b color band values of these pixels. For each (x_p, y_p) $p \in [1, 2, ..., K_p]$ location which satisfy $B_c(x_p, y_p) = 0$ equation, we apply the same procedure and obtain L_p, a_p, b_p values which indicates mean of L, a, b color band values around connected component located at (x_p, y_p) center point. In order to test background similarity, we check if extracted values satisfy inequality given below,

$$\sqrt{(L_c - L_p)^2 + (a_c - a_p)^2 + (b_c - b_p)^2} < \xi \qquad (5)$$

In our study, we select ξ as equal to 10 after extensive tests. Although slight changes of ξ value does not effect detection result, large increase of this threshold might lead to false detections, on the other hand large decrease might lead to inadequate detections. In Fig.2.(c), we provide detected people in non-crowd regions of $Stadium_1$ test image.

3 PEOPLE TRACKING USING KALMAN FILTER

Tracking allows us to see how each person is behaving over time, which provides another very important information. Unfortunately, we cannot track people which appear in dense crowd regions. We try to track individual people in sparse groups that we detected in previous step.

In order to track and extract the motion path of detected people, we benefit from Kalman filter. The Kalman filter is an efficient recursive filter which estimates the state of a dynamic system from a series of incomplete and noisy measurements, developed by Rudolf Kalman (Kalman, 1960). The filtering process consists of two main steps; time update (prediction) step, and measurement update (correction) step. The time update step is responsible for projecting the current state forward in time to obtain the priori estimates for the next time step. The measurement update step deals with the feedback of the system to obtain improved a posteriori values. In our study, we benefit from Kalman filter in order to connect person's positions represented with (x_p^n, y_p^n) for nth image with person's positions represented with (x_p^{n+1}, y_p^{n+1}) in $n + 1$th image. Kalman filter helps us to predict location of a person in the next image of the sequence using the information obtained in the previous image. First, we define the state vector X_n as $X_n = (x_p^n, y_p^n, v_x^n, v_y^n)^T$. Here, (x_p^n, y_p^n), (v_x^n, v_y^n) holds location and velocity vectors of detected people respectively in nth image of the sequence. Observation vector Y_n is defined to represent the location of the people in nth image. The state vector X_n and the observation vector Y_n are related as the following basic system equation;

$$X_n = \phi X_{n-1} + G w_{n-1} \qquad (6)$$

$$Y_n = H X_n + v_n \qquad (7)$$

where ϕ is known as the state transition matrix, G is the driving matrix, H is the observation matrix, and w_n is the system noise added to the velocity of the state vector X_n, and v_n is the observation noise which is the error between real and detected location. As in most of the human motion tracking studies, in our study we assume approximately uniform straight motion for a person in successive images. Then, ϕ, G, H are defined as follows;

$$\phi = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad (8)$$

$$G = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}^T \qquad (9)$$

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \tag{10}$$

Because of the short capturing time between input images, we assume $\Delta t = 1$. w_n and v_n are assumed as constant Gaussian noise with zero mean. Thus the covariance matrix for w_n and v_n become $\sigma_w^2 I_{2\times2}$ and $\sigma_v^2 I_{2\times2}$, where $I_{2\times2}$ represent a 2×2 identity matrix. Finally, we formulate Kalman filter as,

$$K_n = \bar{P}_n H^T (H \bar{P}_n H^T + I_{2\times2})^{-1} \tag{11}$$

$$\bar{x}_p^n = \phi \bar{x}_p^{n-1} + K_{n-1}(y_p^{n-1} - H \bar{x}_p^{n-1}) \tag{12}$$

$$\bar{P}_n = \phi(\bar{P}_{n-1} - K_{n-1} H \bar{P}_{n-1})\phi^T + \frac{\sigma_w^2}{\sigma_v^2} Q_{n-1} \tag{13}$$

where $\bar{x}_p^n$ and $\bar{y}_p^n$ are estimated values of x_p^n and y_p^n. $\bar{P}_n$ equals to C/σ_v^2, C represents the covariance matrix of estimated error of $\bar{x}_p^n$. K_n is Kalman gain, and Q equals to GG^T. Then the predicted location of the feature in $n + 1$th image is given as $(\bar{x}_p(n+1), \bar{y}_p(n+1))$. For more detailed explanation of Kalman filtering process please refer to related reference (Kalman, 1960).

After predicting next movement positions for each person using their previous position information, we use this information for tracking. For jth person detected in $(x_p^n(j), y_p^n(j))$ location in nth image of the sequence, we calculate predicted position in the next step as $(\bar{x}_p^{n+1}, \bar{y}_p^{n+1})$. If we can find a person in $(x_p^{n+1}(l), y_p^{n+1}(l))$ position in the $n+1$th image, where the position satisfies $\sqrt{(\bar{x}_p^{n+1} - x_p^{n+1}(l))^2 - (\bar{y}_p^{n+1} - y_p^{n+1}(l))^2} < 5$ inequality, in another saying if the person in $n + 1$th image in $(x_p^n(j), y_p^n(j))$ location is less than 5 pixels away from the predicted position of the person in $(x_p^n(j), y_p^n(j))$, we assume that $(x_p^{n+1}(l), y_p^{n+1}(l))$ is the next position of this person. Using this information, we can connect motion path of each individual person in input images.

In some cases, because of the low resolution of the airborne images, we cannot detect the same person in each image of the sequence. In that case, for jth person in $(x_p^n(j), y_p^n(j))$ position, we cannot find a $(x_p^{n+1}(l), y_p^{n+1}(l))$ position which satisfies $\sqrt{(\bar{x}_p^{n+1} - x_p^{n+1}(l))^2 - (\bar{y}_p^{n+1} - y_p^{n+1}(l))^2} < 5$ inequality. In order to continue to tracking process, we assume Kalman filter's estimated position $(\bar{x}_p^{n+1}(j), \bar{y}_p^{n+1}(j))$ as the next position of the person. Therefore, besides enabling multiple object tracking, Kalman filter also help us to continue tracking process even if a person cannot be detected in some of the images of the sequence.

In Fig.3 and 4, we provide example tracking results for three different conditions. In Fig.3, in sequence there is also one false person detection in the sequence which does not effect our tracking process. In Fig.4.(a) we present detected motion paths for two persons walking in opposite directions, and in Fig.4.(b) two persons who are walking very close to each other are tracked correctly.

4 EXPERIMENTS

To test our method, we use airborne images which are obtained using a new low-cost airborne frame camera system (named 3K camera) which has been developed at the German Aerospace Center (DLR). The spatial resolution and swath width of the camera system range between $15cm$ to $50cm$, and $2,5km$ to $8km$ respectively. Within two minutes an area of approximately $10km \times 8km$ can be monitored. That high observation coverage gives great advantage to monitor large events. Obtained image data are processed onboard by five computers using data from a real-time GPS/IMU system including direct georeferencing. In this study, our 3K airborne camera image data set consists of a stadium entrance data set ($Stadium_{1-43}$) which includes 43 multi-temporal images. Because of manually focusing difficulties of the current camera system, unfortunately most of the images in our data set are blurred. Although this issue decreases detection capabilities of our system, obtained results can still provide important information about states of the crowds and approximate quantitative measures of crowd and non-crowd regions. Besides, we can apply tracking operation to extract motion path of people in non-crowd regions. We believe that future airborne camera systems with correct focusing capabilities will help us to obtain more accurate estimations.

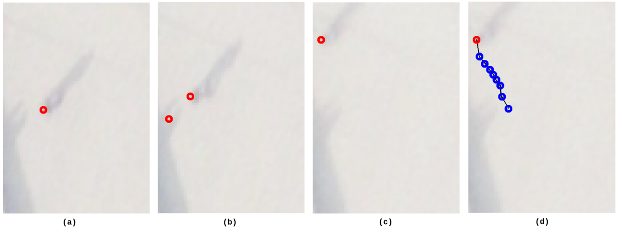
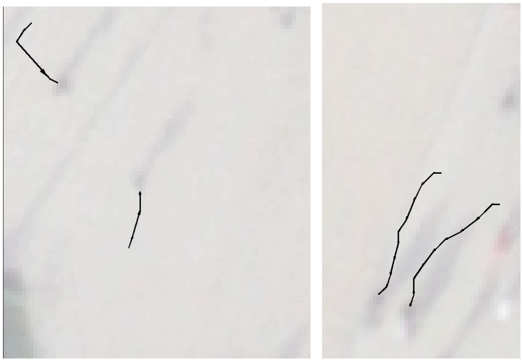


Figure 3: Tracking process of one person. (a) A sub-region from $Stadium_1$ test image, (b) A sub-region from $Stadium_6$ test image (includes also a false detected person location), (c) A sub-region from $Stadium_{15}$ test image, (d) Detected motion path of the person.



(a) (b)

Figure 4: (a) Detected motion paths for two persons walking in opposite directions, (b) Detected motion paths for two persons which are walking very close to each other.

In order to obtain a measure about the performance of crowd analysis step of the algorithm, we have generated groundtruth data for four dense crowds in $Stadium_1$ which are represented in Fig. 5. Since even for human observer it is hard to count the exact number of people in crowds, we have assumed mean of counts of three human observers as groundtruth. In Table 1, we compare automatically detected number of people (N), and density (d) with groundtruth data (N_{gth} and d_{gth} respectively) for each crowd. Similarity of our measures with groundtruth shows the high performance of the proposed approach.

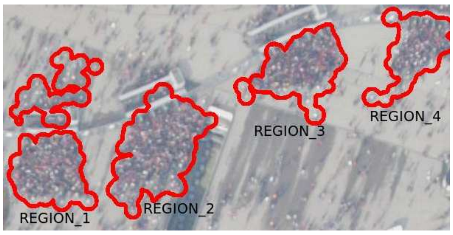


Figure 5: A small part of $Stadium_1$ test image. Labels of detected crowds which are used for performance analysis are written on the image.

Table 1: Comparison of groundtruth and automatically detected people number and density estimation results for test regions in $Stadium_1$. N and d stand for *number of people* and *number of people per square meter* respectively.

	$REGION_1$	$REGION_2$	$REGION_3$	$REGION_4$
N	139	211	115	102
N_{gth}	132	180	114	98
d	0.81	0.74	0.68	0.76
d_{gth}	0.76	0.63	0.67	0.73

In order to measure performance of the tracking step, for each individual person we measure the Euclidean distance between $(\bar{x}_p^n, \bar{y}_p^n)$ position which is estimated at $n-1$th step, with the position (x_p^n, y_p^n) which is detected at nth step. Mean of difference measures which are calculated from each input image of the sequence gives us information about the tracking performance. For jth person of the scene, we calculate the tracking performance with the following equation.

$$Err = \frac{\sum_{n=2}^{N} \sqrt{(x_p^n(j) - \bar{x}_p^n(j))^2 + (y_p^n(j) - \bar{y}_p^n(j))^2}}{N-1} \tag{14}$$

Here, smaller Err value indicates smaller error in the estimations of tracking process, or in other saying smaller Err indicates higher tracking performance. For instance, for the tracking operation represented in Fig.3, Err is calculated as $4,90$. Despite very low image resolutions, 283 of 365 persons in non-crowd regions are detected and tracked automatically.

5 CONCLUSIONS AND FUTURE WORK

In order to solve crowd and people behavior analysis problem, herein we propose a novel and fully automatic approach using airborne images.

Although resolutions of airborne images are not enough to see each person with sharp details, we can still notice a change of color components in the place where a person exists. Therefore, we used local features which are extracted from illumination invariant chroma bands of the image. Assuming extracted local features as observation points, we generated a probability density function using Gaussian kernel functions with constant bandwidth which can adapt itself automatically regarding input image resolutions. Using obtained pdf function, first dense crowd boundaries are robustly detected and quantitative measures are extracted for crowds. For detecting other people, we have detected background color for crowd regions, and we searched for feature locations with similar background color. Detected individual people are tracked in the input image sequence using Kalman filtering. We have tested our algorithm on a stadium entrance airborne image sequence. Our experimental results indicate possible usage of the algorithm in real-life events, also for on-board applications.

REFERENCES

Arandjelovic, O., Sep. 2008. Crowd detection from still images. British Machine Vision Conference (BMVC'08).

Burkert, F., Schmidt, F., Butenuth, M. and Hinz, S., Sep. 2010. People tracking and trajectory interpretation in aerial image sequences. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences (IAPRS), Commission III (Part A) XXXVIII, pp. 209–214.

Davies, A., Yin, J. and Velastin, S., 1995. Crowd monitoring using image processing. IEEE Electronic and Communications Engineering Journal 7 (1), pp. 37–47.

Fairchild, M., 1998. Color appearance models. Addison-Wesley.

Ge, W. and Collins, R., 2009. Marked point process for crowd counting. IEEE Computer Vision and Pattern Recognition Conference (CVPR'09) pp. 2913–2920.

Ge, W. and Collins, R., 2010. Crowd detection with a multiview sampler. European Conference on Computer Vision (ECCV'10).

Hinz, S., 2009. Density and motion estimation of people in crowded environments based on aerial image sequences. ISPRS Hannover Workshop on High-Resolution Earth Imaging for Geospatial Information.

Kalman, R., 1960. A new approach to linear filtering and prediction problems. Journal of Basic Engineering 82 (1), pp. 35–45.

Lin, S., Chen, J. and Chao, H., Nov. 2001. 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), pp. 645–654.

Otsu, N., 2009. A threshold selection method from gray-level histograms. IEEE Transactions on System, Man, and Cybernetics 9 (1), pp. 62–66.

Paschos, G., 2001. Perceptually uniform color spaces for color texture analysis: an empirical evaluation. IEEE Transactions on Image Processing 10, pp. 932–937.

Regazzoni, C. and Tesei, A., 1994. Local density evaluation and tracking of multiple objects from complex image sequences. Proceedings of 20th International Conference on Industrial Electronics, Control and Instrumentation (IECON) 2, pp. 744–748.

Regazzoni, C. and Tesei, A., 1996. Distributed data fusion for real time crowding estimation. Signal Processing 53, pp. 47–63.

Rosten, E., Porter, R. and Drummond, T., Nov. 2010. Faster and better: A machine learning approach to corner detection. IEEE Transactions on Pattern Analysis and Machine Learning 32 (1), pp. 105–119.

Silverman, B., 1986. Density estimation for statistics and data analysis. 1st Edition.

Sirmacek, B. and Unsalan, C., 2010. A probabilistic framework to detect buildings in aerial and satellite images. IEEE Transactions on Geoscience and Remote Sensing.

Sonka, M., Hlavac, V. and Boyle, R., 2007. Image processing: Analysis and machine vision. 3rd Edition Lubbock, TX: CL-Engineering.