Vision-based drone flight control and crowd or riot analysis with efficient color histogram based tracking

Thomas Müller and Markus Müller

Fraunhofer Institute IOSB, Fraunhoferstrasse 1, 76131 Karlsruhe, Germany

ABSTRACT

Object tracking is a direct or indirect key issue in many different military applications like visual surveillance, automatic visual closed-loop control of UAVs (unmanned aerial vehicles) and PTZ-cameras, or in the field of crowd evaluations in order to detect or analyse a riot emergence. Of course, a high robustness is the most important feature of the underlying tracker, but this is hindered significantly the more the tracker needs to have low calculation times. In the UAV application introduced in this paper the tracker has to be extraordinarily quick.

In order to optimize the calculation time and the robustness in combination as far as possible, a highly efficient tracking procedure is presented for the above mentioned application fields which relies on well-known color histograms but uses them in a novel manner. This procedure bases on the calculation of a color weighting vector representing the significances of object colors like a kind of an object's color finger print. Several examples from the above mentioned military applications are shown to demonstrate the practical relevance and the performance of the presented tracking approach.

Keywords: Highly efficient object tracking, color histogram, color finger print, UAVs (unmanned aerial vehicles), automatic visual closed-loop control, crowd evaluation.

1. INTRODUCTION

As one of the main problems in image processing in the military or civil domain, tracking objects in 2D (i.e. in the image plane) or 3D (i.e. in scene space) draws through the history of computer vision in a variety of different approaches, measurement functions and robustness analyses. The main goal is to develop tracking mechanisms which work in as many cases as possible, as robust as possible, and preferably in video real-time, i.e. as fast as the used sensor provides the image data. Since these three aspects are largely contradictory, the tracking problem remains unsolved in general despite the extensive scientific efforts and the power of current computers. These circumstances have led to a development where powerful trackers could be created for some dedicated scenarios.

In this vein, the contribution of this paper relates to the field of 2D tracking applications, where the tracking process has to work notably fast and where the objects to be tracked possess one or more different colors or a different overall colorization in comparison to the background. In this process, the color informations inside and outside the ROI (region of interest; the rectangular box marking the object position in an image) are evaluated and object colors are separated from background colors. Additionally, the object colors are weighted according to their relevance as object characteristics leading to a color weighting vector representing the significances of object colors like a kind of an object's color finger print. In a successive image the object to be tracked is found by an optimization process relying on this object information using an exhaustive search or particle filtering method (leading to the updated ROI's position).

The concluding sections of this paper show the results obtained with the presented tracking approach in different fields of the military domain: in visual surveillance, in the automatic visual closed-loop control of a UAV (unmanned aerial vehicle), and in the field of crowd evaluation in order to detect or analyse a riot emergence.

Further author information:
Thomas Müller: E-mail: Thomas.Mueller@iosb.fraunhofer.de, Telephone: +49(0)721/6091 458.
Markus Müller: E-mail: Markus.Mueller@iosb.fraunhofer.de, Telephone: +49(0)721/6091 250.

2. TRACKING AND MEASUREMENT FUNCTIONS

After the era of Kalman filters and iterative extended Kalman filters (IEKFs) (cf. [1], [2], [3]) which formed a basis framework for tracking algorithms (e.g. a contour- and region-based 3D tracking of persons [4], a contour-based 3D tracking of parts of a car's engine compartment [5] to name examples), particle filters – also known as condensation algorithms [6], [7] – have become very popular in the recent years as a main basis for object tracking (e.g. for contour-based 2D tracking of humans, hands or this well-known bush leaf in the wind [6], [7], [8]).

Since the work of Isard and Blake [7] a couple of improved condensation variants were created, including the mixture particle filter for maintaining multimodality [9], the annealed particle filter [10] for higher-dimensional cases, the Rao-Blackwellisation particle filter [11], [12], [13] for high-dimensional cases, an extension with the so called importance sampling [14], and the kernel density estimation extension [15], [16].

In addition to that, recent and very promising approaches like the one of Porikli, Tuzel and Meer [17] exhaust the fusion of different image data aspects like color, grayvalue gradient, local image features, image texture and others. Compared with previous approaches, these fusion algorithms seem to lift the tracking robustness as well as the tracking capabilities onto a higher level. So far, such algorithms do not work in real-time (especially when having large search regions) which is still the reason to choose real-time trackers with less robustness in a lot of applications (cf. introductory Section 1). Alternatively, in some applications the real-time aspect leads to implementations with a minimum of image data aspects to be fused. This automatically increases the required quality of the fused components. Especially for tracking humans or human's body parts, the color aspect was shown to be an important component for capturing the color of human skin (see, e.g. [18], [19], [20], [21]). Therefore, most fusion papers in the field of tracking humans or human's body parts explore the combination of different image data aspects with color information, especially with color histograms. Kwolek [22], for example, fuses the depth information of a stereo camera, gradient and shape features as well as color histograms to track heads of humans.

The great progress in the field of Kalman and particle filtering as well as fusion methods may have led to an impression that the measurement function has become just a marginal aspect in the tracking context. But all mentioned older and younger approaches rely more or less on the use of an appropriate measurement function with respect to the dedicated tracking problem and, naturally, the tracking robustness depends on the weakest component of the overall tracking system. In the end, a practical reason of using a Kalman or particle filter is to (hopefully) optimally combine the measurements extracted from the image with a suitable object movement model in order to increase the tracking robustness because of weaknesses of the measurement function. But the filter technique will not be able to compensate for a suboptimal measurement function in the long run. In order to optimize the robustness one will have to improve all parts of the tracking algorithm. This paper attends mainly the measurement function in the aspect of color (see Section 3).

An 'optimal' measurement function would be robust enough to be (as much as possible) independent of the use of a movement model in a Kalman or particle filter framework. For example, an exhaustive search procedure (in order not to be stuck in local minima) can do the localization work in every image (see Sections 4, 5 and 8) and provide a robustness reference. So, if this can be done robustly and fast (optimally in real-time) the evaluated quality of the measurement function will directly contribute to a high robustness and a high processing speed when combined with a state-of-the-art Kalman or particle filter (cf. Section 6) or a modern fusion approach. The chosen framework can then just eliminate the remaining robustness problems in more or less accidental situations.

3. COLOR HISTOGRAM APPROACH

3.1 Problem definition

Let

$$i^{(n)} : D \mapsto \{0, \ldots, 255\}^3 \tag{1}$$

with $D := \{0,\dots,w-1\} \times \{0,\dots,h-1\}$ be the n-th color image in an image sequence of size $w \times h$ pixels with $i^{(n)}(x,y) = (c_1^{(n)}, c_2^{(n)}, c_3^{(n)})^T$ representing the three color channels at image position $(x,y)^T$ with respect to a chosen color space model, for example RGB, HSI or LAB.*

Furthermore, let

$$T^{(n)} := \{l^{(n)},\dots,r^{(n)}\} \times \{u^{(n)},\dots,d^{(n)}\} \subset D \tag{2}$$

be the ROI, i.e. a rectangular tracking region in image $i^{(n)}$ with the horizontal, i.e. left and right borders $l^{(n)}$, $r^{(n)}$ and the vertical borders $u^{(n)}$ and $d^{(n)}$. In this paper, $T^{(1)}$ is assumed to be given by an object specific detection algorithm or is defined manually for the first image $i^{(1)}$ by the user, for example by drawing a rectangle with the mouse around the interesting object to be tracked. This rectangle may be greater than minimally needed and surrounds, naturally, not only the object structure but also pixels steming from the scene background.

The tracking task can be formulated as follows: calculate the tracking region $T^{(n+1)}$ with respect to image $i^{(n+1)}$ out of $i^{(n+1)}$, $T^{(n)}$ and $i^{(n)}$ for $n = 1, 2, \dots$ so that the bounding box around $T^{(n+1)}$ surrounds the object in image $i^{(n+1)}$.

3.2 Color reduction and histogram calculation

As a first step of our proposed measurement function, we apply a color reduction function

$$f : \{0,\dots,255\}^3 \mapsto \{0,\dots,2^k-1\}^3 \tag{3}$$

on image $i^{(n)}$, where $k < 8$ denotes the number of target color bits. On the one hand, the color reduction leads directly to advantages concerning an algorithmic differentiation between relevant and irrelevant neighbourhood relations which is necessary due to color noise in the images and plenty of other color effects like shading, illumination changes, emerging mixed colors and so forth. On the other hand, the tracking calculations can be done efficiently with a reduced color number. Our optimization experiments led us to the value of $k = 4$ as an appropriate value with respect to the tracking results as well as the processing time. This performs a reduction from all the $256^3 \approx 16.8$ million colors (8 bit per color channel) to $(2^k)^3 = 2^{12} = 4096$ colors (4 bit per channel). This step can be done very easily and efficiently with an arithmetic logical shift of the color bytes 4 bit positions to the right which is equivalent to a division by 16 and truncating the decimal places, i.e. $f(c_1,c_2,c_3) = (\lfloor \frac{c_1}{16} \rfloor, \lfloor \frac{c_2}{16} \rfloor, \lfloor \frac{c_3}{16} \rfloor)^T$, $c_1,c_2,c_3 \in \{0,\dots,255\}$.†

For a set $M \subseteq D$ of image pixels the function

$$\begin{aligned} H_M^{(n)} &: \{0,\dots,2^{3k}-1\} \mapsto I\!N_0 \ , \\ H_M^{(n)}(i) &:= |\left\{(x,y) \in M \mid p(f(i^{(n)}(x,y))) = i\right\}| \end{aligned} \tag{4}$$

is called the *color histogram* of image $i^{(n)}$ with respect to M. The argument i is a color index and the function p calculates this index out of a (reduced) color $(c_1,c_2,c_3)^T$:

$$\begin{aligned} p &: \{0,\dots,2^k-1\}^3 \mapsto \{0,\dots,2^{3k}-1\} \ , \\ p(c_1,c_2,c_3) &:= 2^{2k}c_1 + 2^k c_2 + c_3 \ . \end{aligned} \tag{5}$$

In the next step of the algorithm, the histogram $H_{T^{(n)}}^{(n)}$ for the tracking region $T^{(n)}$ and the histogram $H_{B^{(n)}}^{(n)}$ for the background region

$$B^{(n)} := \left(\left(\{l^{(n)}-b,\dots,r^{(n)}+b\} \times \{u^{(n)}-b,\dots,d^{(n)}+b\}\right) \setminus T^{(n)}\right) \cap D \tag{6}$$

are calculated. b is a constant parameter for the size of the background region, i.e. the area around the tracking region. It should be equal to or greater than the maximal expected movement of the object from one image to the next one‡.

*Interestingly, against our expectations, our experiments have shown no significant advantage in using HSI or LAB instead of RGB. Therefore, we work with the RGB model because of processing speed.

†Of course, f can be calculated alternatively with a more powerful color reduction function, if a higher processing time is uncrucial. But this issue shall not be a subject in this paper.

‡for example $b = 50$ or $b = 100$ [pixels].

3.3 Rating of characteristic object colors via color weighting

A key issue of the presented approach is the calculation of characteristic colors via a color ranking $w^{(n)}$ for all possible colors. This function determines the significance of each (reduced) color $(c_1, c_2, c_3)^T \in \{0, \ldots, 2^k - 1\}^3$, identified with $i = p(c_1, c_2, c_3)$, with respect to the tracking problem according to

$$w^{(n)}(i) := \begin{cases} w_1, & \text{if } H^{(n)}_{T^{(n)}}(i) > H^{(n)}_{B^{(n)}}(i) = 0 \\ w_2, & \text{if } H^{(n)}_{T^{(n)}}(i) > H^{(n)}_{B^{(n)}}(i) > 0 \\ w_3, & \text{if } H^{(n)}_{T^{(n)}}(i) \leq H^{(n)}_{B^{(n)}}(i) \\ 0, & \text{if } H^{(n)}_{T^{(n)}}(i) = 0 \end{cases} , \tag{7}$$

where

$$w_1 := \min\left\{ H^{(n)}_{T^{(n)}}(i) \sum_{v=0}^{3} gauss(v)\, e_1(v),\ w_{\max} \right\} , \tag{8}$$

$$w_2 := \min\left\{ \frac{H^{(n)}_{T^{(n)}}(i)}{4 H^{(n)}_{B^{(n)}}(i)} \sum_{v=0}^{3} gauss(v)\, e_2(v),\ w_{\max} \right\} , \tag{9}$$

$$w_3 := \frac{H^{(n)}_{T^{(n)}}(i)}{H^{(n)}_{B^{(n)}}(i)} \tag{10}$$

with $e_1(v)$, $e_2(v)$, $gauss(v)$, and $w_{\max}$ as follows.

$e_1(v)$ denotes the number of neighbour colors of $(c_1, c_2, c_3)^T$ in distance v for which $H^{(n)}_{B^{(n)}}(j) = 0$ holds and $e_2(v)$ the number of neighbour colors for which $H^{(n)}_{T^{(n)}}(j) > H^{(n)}_{B^{(n)}}(j)$ holds:

$$e_1(v) := \left| \left\{ j \;\middle|\; H^{(n)}_{B^{(n)}}(j) = 0,\ j = p\left(\begin{pmatrix} c_1 + \Delta c_1 \\ c_2 + \Delta c_2 \\ c_3 + \Delta c_3 \end{pmatrix} \right), \begin{pmatrix} \Delta c_1 \\ \Delta c_2 \\ \Delta c_3 \end{pmatrix} \in \{-1, 0, 1\}^3,\ \left\| \begin{pmatrix} \Delta c_1 \\ \Delta c_2 \\ \Delta c_3 \end{pmatrix} \right\| = v \right\} \right| , \tag{11}$$

$$e_2(v) := \left| \left\{ j \;\middle|\; H^{(n)}_{T^{(n)}}(j) > H^{(n)}_{B^{(n)}}(j),\ j = p\left(\begin{pmatrix} c_1 + \Delta c_1 \\ c_2 + \Delta c_2 \\ c_3 + \Delta c_3 \end{pmatrix} \right), \begin{pmatrix} \Delta c_1 \\ \Delta c_2 \\ \Delta c_3 \end{pmatrix} \in \{-1, 0, 1\}^3,\ \left\| \begin{pmatrix} \Delta c_1 \\ \Delta c_2 \\ \Delta c_3 \end{pmatrix} \right\| = v \right\} \right| \tag{12}$$

with the vector norm $\|(x_1, x_2, x_3)^T\| = |x_1| + |x_2| + |x_3|$.

$gauss(v)$ is proportional to a discretization of the gaussian function

$$g(x) := \frac{1}{\sqrt{2\pi}\sigma} \cdot e^{-\frac{1}{2}(\frac{x-\mu}{\sigma})^2} \tag{13}$$

having the maximum $gauss(0) = 1$.

$w_{\max}$ denotes a fixed parameter delimiting the contribution of single colors to the measurement function. Experiments have shown that it is beneficial to set this parameter to an order of magnitude greater than 1, for example $w_{\max} = 20$, since $w_3 < 1$. If the tracked object possesses useful characteristic colors with respect to the examined background, w_1 and w_2 will lead to significant contributions of the associated colors and the contribution of w_3 will be nearly neglected (see the following definition of the measurement function). Otherwise, if the colorization of the object is disadvantageous, w_1 and w_2 do not contribute and w_3 will try to make the best out of the bad color situation.

3.4 Measurement function

After calculating $H^{(n)}_{T^{(n)}}$ and the color weighting $w^{(n)}$ with respect to image $i^{(n)}$, the presence of the object in a region of image $i^{(n+1)}$ can be evaluated as follows.

Let

$$X^{(n+1)}(\Delta x, \Delta y) := \{l^{(n)}+\Delta x, \ldots, r^{(n)}+\Delta x\} \times \{u^{(n)}+\Delta y, \ldots, d^{(n)}+\Delta y\} \tag{14}$$

be a displacement of the tracking region $T^{(n)}$ with parameters $\Delta x, \Delta y$ so that $X^{(n+1)}(\Delta x, \Delta y) \subset D$.

We define the figure of merit for $X^{(n+1)}(\Delta x, \Delta y)$ as

$$h^{(n+1)}(\Delta x, \Delta y) := \sum_{i=0}^{2^{3k}-1} w^{(n)}(i) \cdot \min\left\{H^{(n)}_{T^{(n)}}(i),\ H^{(n+1)}_{X^{(n+1)}(\Delta x, \Delta y)}(i)\right\} \ . \tag{15}$$

4. REAL-TIME TRACKING VIA EXHAUSTIVE SEARCH

The tracking problem can be solved by calculating $h^{(n+1)}(\Delta x, \Delta y)$ for every $(\Delta x, \Delta y)^T \in \{-m, \ldots, m\}^2$ for a predefined $m \in \mathbb{N}$ and maximizing the figure of merit

$$(\Delta x_{\max}, \Delta y_{\max})^T := \arg \max_{\substack{(\Delta x, \Delta y)^T \in \{-m, \ldots, m\}^2, \\ X^{(n+1)}(\Delta x, \Delta y) \subset D}} h^{(n+1)}(\Delta x, \Delta y) \ , \tag{16}$$

$$T^{(n+1)} := X^{(n+1)}(\Delta x_{\max}, \Delta y_{\max}) \ . \tag{17}$$

Note that in order to enable tracking, m has to be equal to or greater than the maximal possible movement of the object in pixels from one image to the next one. Parameter b (see end of Subsection 3.2) is less crucial than m, it can just (but doesn't need to) be chosen as $b = m$. Of course, for very dynamic object and/or camera movements, m can be chosen so that the algorithm looks for the object in the whole image $i^{(n+1)}$.

The described exhaustive search for the object in the movement area $\{-m, \ldots, m\}^2$ can be implemented with an efficient runtime. Since the figure of merit only depends on histograms of image $i^{(n+1)}$, the histogram needs only to be calculated in total for the first value of $(\Delta x, \Delta y)^T$, for example $(\Delta x, \Delta y)^T = (-m, -m)^T$, and it can then successively be modified by changing $(\Delta x, \Delta y)^T$ by one pixel steps in the first or the second component until the whole movement area $\{-m, \ldots, m\}^2$ is covered. In each step, the previously calculated histogram needs just to be modified in two horizontal or vertical border lines of $X^{(n+1)}$, one line which is added and one line which is removed.

5. INCORPORATING SHAPE AND SIZE CHANGES OF THE OBJECT

After the tracking step, the color weighting information $w^{(n)}$ can be used to adjust the borders $l^{(n+1)}$, $r^{(n+1)}$, $u^{(n+1)}$ and $d^{(n+1)}$ of $T^{(n+1)}$ according to the colors in image $i^{(n+1)}$. Iteratively for the four borders, the image colors of $i^{(n+1)}$ are evaluated perpendicular to the border direction. In this evaluation the colors with a weight exceeding a given threshold are counted which leads to a function describing the number of significant colors perpendicular to the border line's direction, i.e. on the border line as well as on its both sides. An analysis of this function in relation to the size of the tracking region leads to the decision if the border line is moved, to which direction and by how many pixels is it moved perpendicular to the border line's direction.

For illustration, a tracking example is shown in Figure 1 which incorporates shape and size changes. The figure shows the tracking of a yellow pennon at the left roadside. The image sequence was recorded with a camera in a car driving at a speed of about 65 km/h. The figure shows every sixth image, i.e. the depicted images are recorded with a time difference of $6 \cdot 40$ milliseconds $= 240$ milliseconds. Further 240 milliseconds later, the pennon has passed and therefore isn't visible in the camera images any more. As one can see, the growth of the object in the image is significant but can be handled without problems. Figure 2 illustrates the values of the weighting function $w^{(n)}$ applied on the colors of image $i^{(n+1)}$ in $T^{(n+1)}$ graphically for the two lower results in Figure 1. It shows that the background colors are clearly faded out by the tracking process due to construction

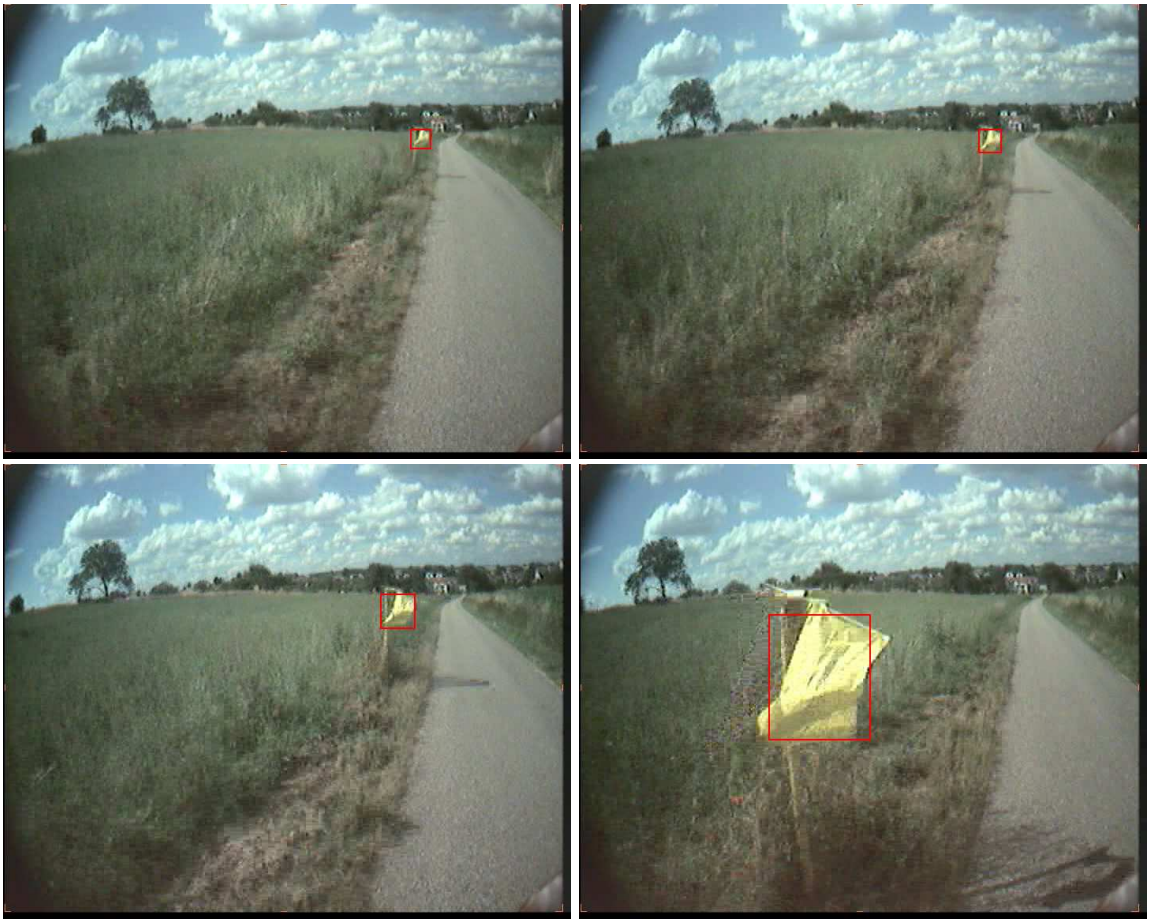

Figure 1. Tracking a yellow pennon at the left roadside while driving with a car at a speed of about 65 km/h.

of the figure of merit. The figure shows the situations before the appliance step of the shape and size tuning process described above and illustrates the mentioned relation between the border lines and the number of high weighted colors. Metaphorically speaking, the borders are moved in a way that most of the white colored informations in Figure 2 are inside the ROI $T^{(n+1)}$ under the constraint that the ROI should be as small as possible.

6. OBJECT TRACKING WITH A PARTICLE FILTER

The presented figure of merit can easily be used as the measurement function of a particle filter. The 2D particles correspond to the arguments $(\Delta x, \Delta y)^T$ of $X^{(n+1)}$. Each particle is evaluated according to the figure of merit $h^{(n+1)}(\Delta x, \Delta y)$. After processing all particles, each evaluation is normalized with the sum of all evaluations in order to get a density distribution. The position $(\Delta x_{\max}, \Delta y_{\max})^T$ of the object, i.e. the argument of $X^{(n+1)}$ in order to get $T^{(n+1)}$ according to equation (17), is calculated as the expectation value of the particles.

7. COMPARISON OF EXHAUSTIVE SEARCH AND PARTICLE FILTER TRACKER

Experiments have shown that in a lot of cases the exhaustive search and the particle filter variant lead to a more or less similar robustness. But there are differences in some situations.

Our implementation of the particle filter works more robust than the exhaustive search variant in a few situations where the object moves slowly and more or less continuously over an image sequence. As expected at the end of Section 2 the particle filter variant compensates for weaknesses in accidental situations there.

Figure 2. Visualisation of the weighting function $w^{(n)}$ for the two lower tracking results in Figure 1. $w^{(n)}$ was applied on the colors of image $i^{(n+1)}$ and displayed via overlayed grayvalues. The darker the pixels in the tracking region, the smaller is the color weight $w^{(n)}$. And the brighter they are, the more characteristic could the respective image colors be categorized.

However, we also observed situations where the exhaustive search variant has shown better results than the particle filter tracker. Especially (but not only) when the camera and/or the objects show a high movement dynamic or speed (leading to a large search region $\{-m,\ldots,m\}^2$), the particle filter loses an object while the exhaustive search method is robust. This seems to be connected with the combination of the movement model and the way the movement noise is handled in the particle filter. So far, we couldn't eliminate this weakness of the particle filter tracker with parameter or algorithmical modifications.

Therefore, we prefer the exhaustive search variant which seems to work better in general. In some situations with slower and more continuous object movements we switch to the particle filter variant.

8. RESULTS IN THE MILITARY APPLICATION DOMAIN

In this section the results are presented in the military applications that are mentioned at the beginning of this paper. All depicted results were calculated by applying the exhaustive search method. The calculation times vary (depending on the size of the ROI and the needed size of the search region $\{-m,\ldots,m\}^2$) from 10 to 20 milliseconds per image on an Intel Pentium D PC with 3.2 GHz (including result visualization). The processed images are 720×576 pixels in size (except for the sequence in Figure 8 with 752×480 pixels).

8.1 Visual Surveillance

In Figure 3 a suspicious running person is tracked robustly in a visual-optical sequence of a camera which is mounted on a remote controlled security quadrocopter. The tracking works robustly despite several disadvantageous image artefacts due to an analog radio connection which transmits the image data to the processing computer on the ground.

Figure 4 shows a similar situation where the visual-optical sensor on the quadrocopter is replaced by a thermal infrared camera. This example documents that the presented method is, of course, not restricted to color images. Actually, persons and vehicles often exhibit quite characteristic grayvalue signatures which can be captured well by the presented tracking approach.

In Figure 5 an observing, possibly spying or even potentially attacking UAV is observed by a ground mounted visual-optical camera and tracked robustly for security purposes. Note that the tracking runs successfully while the drone is dipping from the sky-blue background into the dark green forest background. This ability of the tracking algorithm is a main strength of the presented algorithm due to the construction of the figure of merit. The last image in Figure 5 documents a successful tracking in presence of the brightness correction of the recording camera.

Figure 3. Tracking a suspicious running person with a visual-optical camera mounted on a security quadrocopter. Four single images are picked out of the sequence and depicted here to represent the tracking results for the sequence.

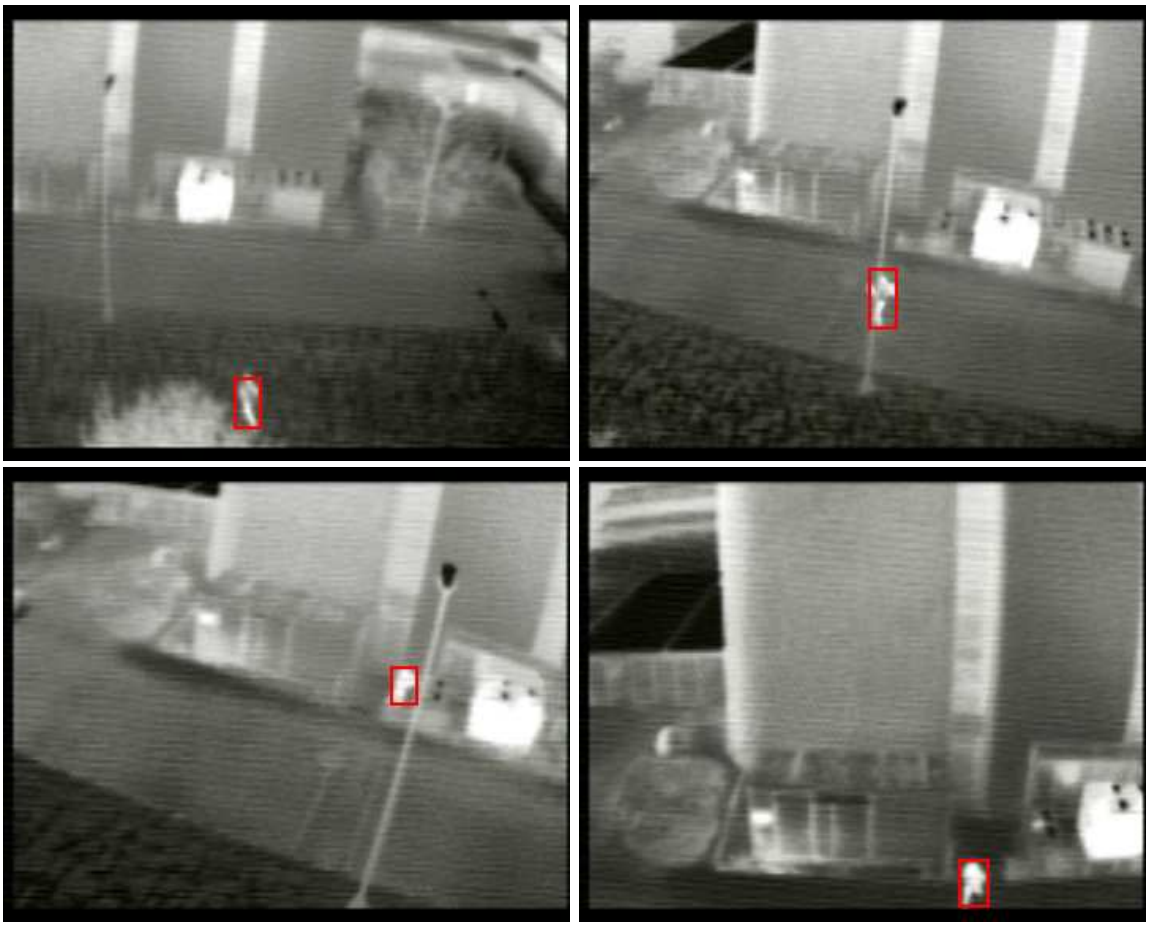

Figure 4. Tracking a suspicious running person with a thermal infrared camera mounted on a security quadrocopter.

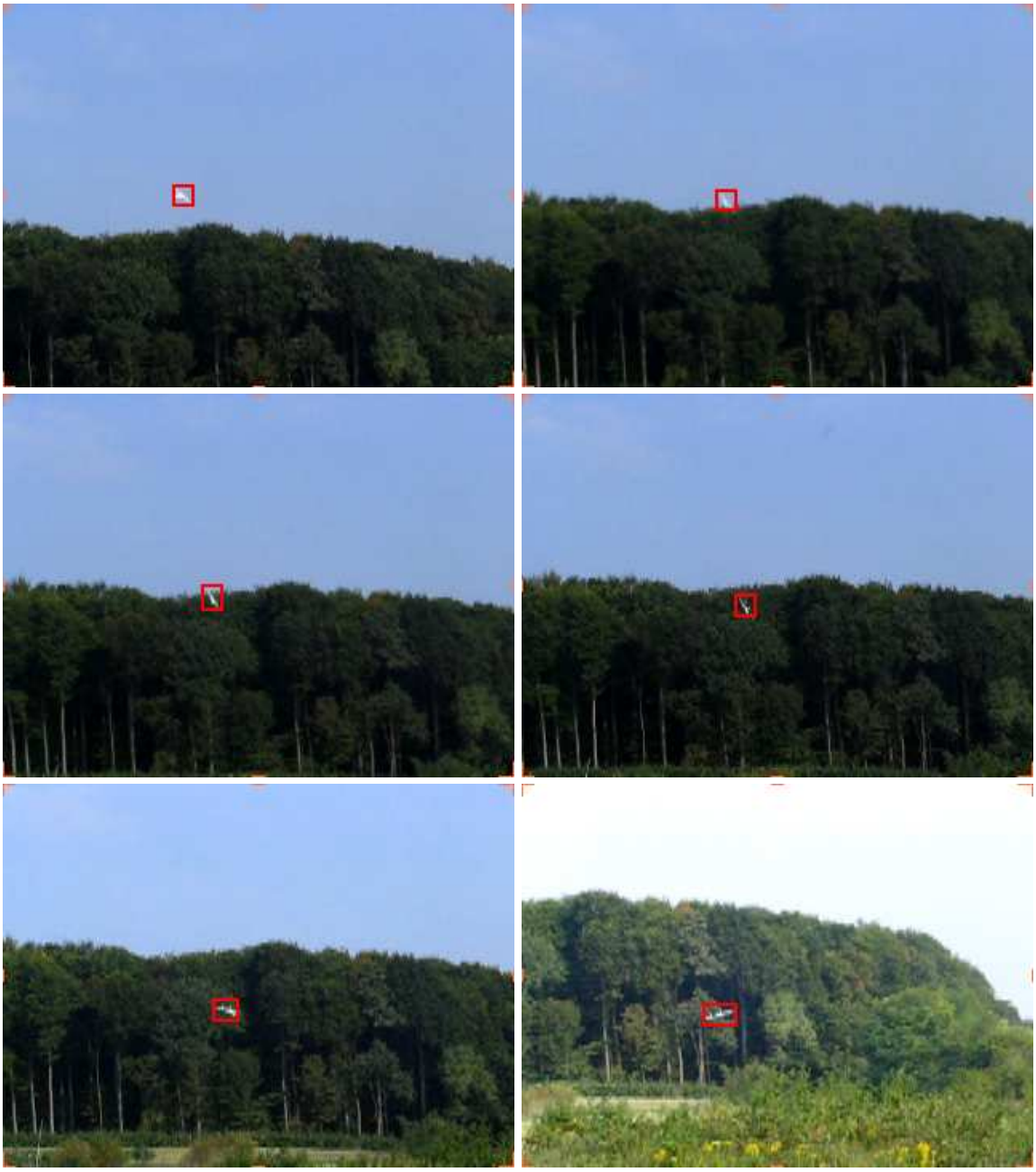

Figure 5. Tracking an intruding UAV using a visual-optical camera.

8.2 Automatic Visual Closed-Loop Control of a UAV

The probably most challenging military application mentioned here is the following online closed-loop control of a flying UAV. The drone is an airplane type flying object with pitch elevators that are controlled by the tracking process. After manually marking a desired target object in the image data acquired by the on board mini camera the task is to track the object in order to fully automatically control the flight path towards the (possibly moving) target object. In this way, the target object can be affected by triggering a certain non-lethal on board instrument as soon as a dedicated object distance is reached. For example, a terrorist can be catched by shooting a carried trapping net over him.

Figure 6. Tracking a target on a grassland with a UAV using an analog on board camera using PAL resolution (in the control loop PAL/4 is used, see text).

Figure 6 depicts a tracking example of the UAV application with a person-like target on a grassland. On the one hand, the control loop has to react rapidly when the target object is moving towards an image boundary in order not to loose the object because of moving out of the image plane. On the other hand, the control has to be quick due to the high flying speed of such an airplane drone especially when the target object approaches to near view positions. This leads to a high dynamic in image changes and in the object's position in the image plane which, by the way, messes up the use of motion models completely. Therefore, the challenge is to work with large search regions in the tracking process in addition to small reaction and tracking times. In order to fulfill the reaction time constraint the tracking process works on-line with 50 Hz image acquisition of PAL/4 images (i.e. images of 360×288 pixels in size) in the control loop.[§]

[§]Since 50 Hz image acquisition of PAL/4 leads directly to an improvement of the reaction time of factor 2 in comparison to 25 Hz PAL images.

For each image the calculated object position is discretized to one of 15 horizontal times 15 vertical image tiles and the tile number is packed into a byte which is sent to the pitch elevator servos via serial interface. The upper half of this byte defines the vertical deviation from the center tile and the lower byte half the horizontal deviation. Figure 7 illustrates this mechanism. The more distant the object is from the image center the more the pitch elevators correct the flight path.

Figure 7. Generating the information for the pitch elevator servos in the upper left situation in Figure 6.

8.3 Crowd or Riot Evaluation

Figure 8 depicts a crowd scenario in which the emergence of a riot situation has to be examined in order to find out how riot scenarios can be detected automatically. The first step in this task is, of course, tracking of some individuals of interest in order to be able to evaluate trajectories, movement dynamics and so on. In the depicted example tracking was done in parallel for seven different persons over an interesting sequence segment of 220 images in length. Tracking time is an important factor in this application domain when having big crowds where lots of persons have to be tracked in video real-time.

9. CONCLUSIONS AND FUTURE WORK

An efficient color-based figure of merit for object tracking was presented as well as two tracking variants using it. The exhaustive search variant has shown as powerful, fast and robust in a lot of cases. By the use of a particle filter framework the figure of merit could be fused easily with an object movement model. The benefit of the presented methods in military application domains could be demonstrated by several examples. Especially when calculation time is a critical resource the presented tracking algorithms serve the niche when minimal calculation times are required in combination with a quite high tracking robustness.

Experimental series in long-time tests could be done now, especially with published computer vision test and benchmark image sequences in order to document the robustness quantitatively via generated statistics. Furthermore, a quantitative comparison with other published trackers and figure of merit functions would be interesting.

Of course, more calculation time may allow more complex and therefore more robust tracking algorithms. Due to simplicity of the figure of merit (which bases only on color histograms to make it fast, abstracting from color positions and structure) the robustness of the presented tracker can significantly be limited in situations where the object to be tracked exhibits not enough color characteristics with respect to the background. In such cases, if calculation time is critical, future work will deal with algorithmic improvements of the methods presented here. If more calculating time is available, the presented figure of merit can easily be combined with a fusion algorithm (cf. Section 2) contributing to a higher overall robustness.

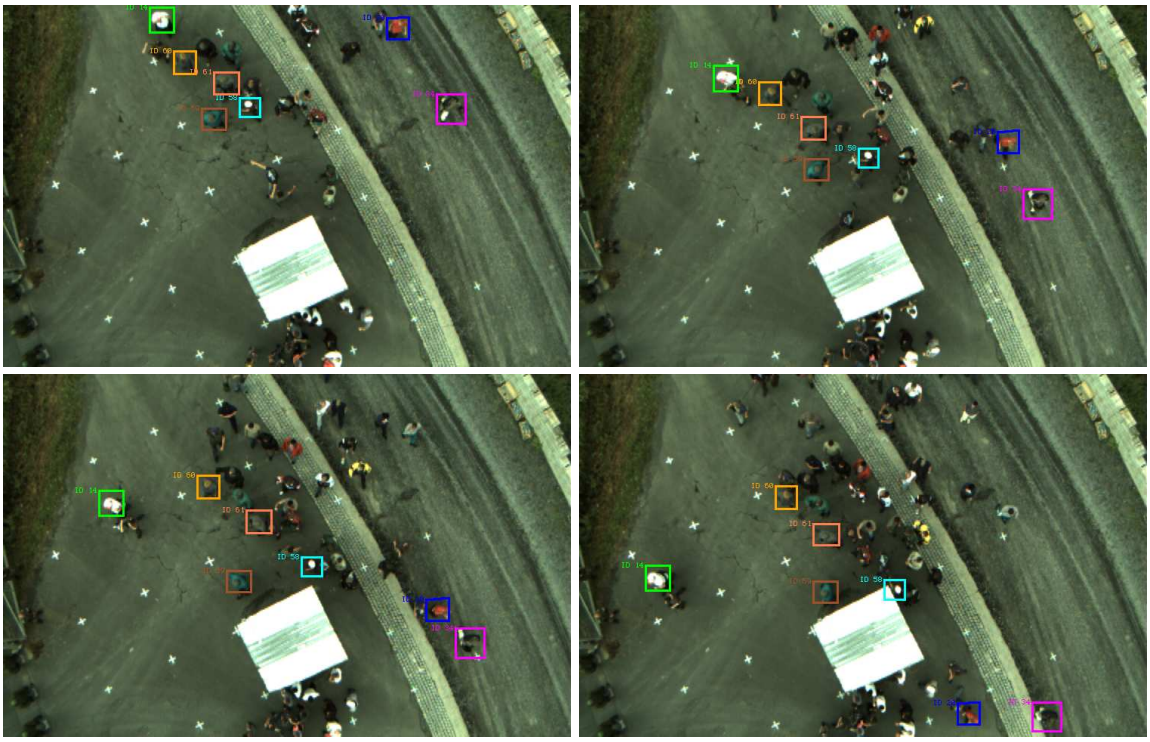

Figure 8. Tracking some persons in a crowd for riot scenario examinations.

ACKNOWLEDGMENTS

The authors gracefully acknowledge the contribution of Sebastian Zettl of the Fraunhofer ICT in Karlsruhe in building up the UAV hardware.

REFERENCES

[1] A. Gelb, J. F. Kasper, R. A. Nash, C. F. Price, A. A. Sutherland: *Applied Optimal Estimation*. MIT Press, Cambridge, MA, USA, 1974.

[2] Y. Bar-Shalom, T. E. Fortmann: *Tracking and Data Association*. Academic Press, San Diego, CA, USA, 1988.

[3] M. S. Grewal, A. P. Andrews: *Kalman Filtering, Theory and Practice*. Prentice Hall, Englewood Cliffs, New Jersey, USA, 1993.

[4] S. Wachter, H.-H. Nagel: *Tracking Persons in Monocular Image Sequences*. Computer Vision and Image Understanding 74:3, 1999, pp. 174–192.

[5] M. Tonko, J. Schürmann, K. Schäfer, and H.-H. Nagel: *Visually Servoed Gripping of a Used Car Battery*. In C. Laugier (Ed.), Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, Grenoble, France, 7–11 September 1997. INRIA: Domaine de Voluceau B.P. 105, Le Chesnay, France, pp. 49–54.

[6] M. Isard: *Visual Motion Analysis by Probabilistic Propagation of Conditional Density*. PhD thesis, Robotics Research Group, Department of Engineering Science, University of Oxford, 1998.

[7] M. Isard and A. Blake: *Condensation – conditional density propagation for visual tracking*. International Journal of Computer Vision 29(1), 1998, pp. 5–28.

[8] The Condensation Algorithm Home Page: `http:// www.robots.ox.ac.uk/~misard/condensation.html` .

[9] J. Vermaak, A. Doucet, and P. Pérez: *Maintaining Multimodality through Mixture Tracking*. In Proceedings of the ninth IEEE International Conference on Computer Vision, 2003.

[10] J. Deutscher, A. Blake, and I. Reid: *Articulated Body Motion Capture by Annealed Particle Filtering*. In Computer Vision and Pattern Recognition, Volume 2, Hilton Head Island, South Carolina, USA, 13–15 June 2000, pp. 126–133.

[11] G. Casella, and C. P. Robert: *Rao-Blackwellisation of Sampling Schemes*. Biometrika 83(1), 1996, pp. 81–94.

[12] A. Doucet: *On Sequential Simulation-based Methods for Bayesian Filtering*. Technical Report CUED/F-INFENG/TR 310, Department on Engineering, Cambridge University, 1998.

[13] A. Doucet, N. de Freitas, K. Murphy, and S. Russell: *Rao-Blackwellised Particle Filtering for Dynamic Bayesian Networks*. Uncertainty in Artificial Intelligence, 2000.

[14] B. Ripley: *Stochastic Simulation*. New York, Wiley, 1987.

[15] C. Cheng, and R. Ansari: *Kernel Particle Filter: Iterative Sampling for Efficient Visual Tracking*. International Conference on Image Processing, Volume 2, 14-17 September 2003, pp. 425–435.

[16] B. W. Silverman: *Density Estimation for Statistics and Data Analysis*. London: Chapman and Hall, 1986.

[17] F. Porikli, O. Tuzel, P. Meer: *Covariance Tracking Using Model Update Based on Lie Algebra*. IEEE Computer Society Conference on Computer Vision and Pattern Recognition CVPR, Volume 1, June 2006, pp. 728–735.

[18] S. Birchfield: *Elliptical Head Tracking Using Intensity Gradients and Color Histograms*. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition CVPR, 1998, pp. 232–237.

[19] G. R. Bradski: *Computer Vision Face Tracking For Use in a Perceptual User Interface*. Intel Technology Journal (2), 1998.

[20] C. R. Wren, A. J. Azarbayejani, T. Darrell, A. P. Pentland: *Pfinder: Real-Time Tracking of the Human Body*. IEEE Transactions on Pattern Analysis and Machine Intelligence 19 (7), July 1997, pp. 780–785.

[21] S. J. McKenna, S. Gong: *Tracking Faces*. In Proceedings of the Second International Conference on Automatic Face and Gesture Recognition, Vermont, USA, 1996, pp. 271-277.

[22] B. Kwolek: *Stereovision-Based Head Tracking Using Color and Ellipse Fitting in a Particle Filter*. European Conference on Computer Vision, Prague, Czech Republic, Lecture Notes in Computer Science, Springer Verlag, Vol. 3024, 2004, pp. 192–204.