Preprints of the 21st IFAC World Congress (Virtual)
Berlin, Germany, July 12-17, 2020

Detection, Localization and Volume Estimation of Deflagrations

Jakob Krooß * **Felix Kümmerlen** ** **Prof. Alexander Fay** *

* *Department of Automation Technology, University of the Federal Armed Forces Hamburg, Germany (e-mail: jakob.krooss@hsu-hh.de, alexander.fay@hsu-hh.de).*
** *Bundeswehr Research Institute for Protective Technologies and NBC Protection (WIS) Munster, Germany (e-mail: FelixKuemmerlen@bundeswehr.org)*

Abstract: Deflagration-like combustions pose a serious threat in various (e.g. industrial) scenarios. Detecting, locating and distinguishing these in their developing phase may significantly reduce the resulting damage to material and individuals. In this work, an image processing based algorithm for detection, localization and volume estimation of deflagrations with a multi-camera-system is proposed. The proposed algorithm has been tested with image sequences of real deflagrations as well as possible false alarm scenarios. In comparison to state of the art methods, false alarm safety, localization quality and robustness to noise have been improved significantly.

Keywords: alarm systems, classification, deflagrations, detection algorithms, fire protection, image processing, image segmentation, multiple view geometry

1. INTRODUCTION

Deflagrations may occur when combustible material in the form of gas, small droplets or dust mixes with an oxidant, usually oxygen (VDI 2263). In such cases, the oxidant is available in the whole volume which leads to an explosion-like combustion when ignited. Especially in closed spaces, deflagrations may do big harm to people and material, however, one may try to detect and extinguish a deflagration in its early development to minimize effects to the surroundings.
With prohibition of halon agents due to its negative effect to environment, alternative systems for extinction (like water mist systems) have to be used (Kim et al. (2004)). These need, in addition to the trigger event itself, the position of a potential deflagration to reach sufficient efficiency during extinction. Hence, in previous works at the Department of Automation Technology, an algorithm has been developed to detect and locate deflagrations using multiple high-speed RGB-cameras (Schröder et al. (2013); Kümmerlen et al. (2013); Schröder et al. (2014); Ernst et al. (2016)). In recent years, nearly every part of the algorithm has been improved significantly.

In the following, the current state of the art is discussed. Subsequently, the most recent version of the proposed algorithm is presented. It is then evaluated with image sequences of real deflagrations as well as possible false alarm scenarios. Finally, the algorithm and its improvements to the state of the art are discussed.

2. STATE OF THE ART AND RELATED RESEARCH

The detection of deflagrations has to be clearly distinguished from detection of fire/flames. Fire detection systems use a wide variety of sensors for smoke, temperature, pressure and light. Only sensors for pressure and light are capable of detecting deflagrations in time. Pressure sensors are sensitive to false alarms and can thus only be used reliably under certain conditions. E.g., they are common in industrial applications (VDI 2263). Light sensors for IR- and/or UV-radiation are used for example for deflagration detection in armored vehicles. In these, a deflagration has to be detected within approximately 15 ms to ensure timely extinction. Under good circumstances, IR/UV-sensors are fast enough to detect a deflagration in 5 ms or less.
However, all of the listed sensors share one problem: A reasonable localization of a fire or deflagration is not possible. As a result, a wide field of research covers the detection and/or segmentation of fire/flames using the images of surveillance cameras or similar. Main advantage is the possibility to extract further information about the combustion, especially its location. An overview is given in Cetin (2013). Algorithms are, amongst others, based on spectral characteristics, an analysis of flickering frequencies and/or a background model. In recent studies, mainly algorithms based on neural networks are proposed, e.g. Muhammad et al. (2018). Qi et al. (2018) described a basic algorithm for classification of deflagrations in closed vessels.
Most of the aforementioned algorithms are either to slow or due to other reasons inapplicable for real-time detection and segmentation of rapid combustion processes like deflagrations. E.g., the rather simple algorithm in Qi et al. (2018) does not consider false alarm scenarios at all. Detection of deflagrations using a neural network is not realistic due to the poor availability of training data. However, the algorithm proposed in this work does use basic elements

of fire detection and segmentation algorithms in literature, as stated at the respective passages later.
Due to the shortcomings of available algorithms, in previous works (Schröder et al. (2013); Kümmerlen et al. (2013); Schröder et al. (2014); Ernst et al. (2016)), a novel algorithm to detect deflagrations in the images of high-speed RGB-cameras has been developed. A main focus of these works is to minimize computational cost to allow for real-time surveillance.
In addition, the use of a multi-camera-system allows for a localization of the deflagration, using the moments of the deflagrations segmentations. The algorithm for three-dimensional localization proposed in Ernst et al. (2016) is limited to two cameras (with the possibility to average the results of multiple camera pairs when data from more cameras is available). It is also, like many algorithms in the field of multiple-view geometry, highly vulnerable to outliers, which are likely to occur in deflagration localization due to reflections or occlusions.

In the following the most recent algorithm for deflagration detection, localization and volume estimation is presented in its whole. It is based on the aforementioned state of the art methods and addresses multiple of their shortcomings. Main contributions of this work are an additional segmentation for localization, a better suited, more robust algorithm for 3D-localization and a lightweight algorithm for volume estimation. Other parts of the state of the art algorithm benefit from these additions as well, e.g. information from localization is now used to remove the distance-dependence of sensitivity during detection and information of the localization segmentation gives additional false alarm security.
The algorithm used for localization in this work is based on a cost function that differs from the L_2 (Hartley and Sturm (1997)) and L_∞ (Hartley and Schaffalitzky (2004)) cost functions used typically in triangulation to address the severe presence of outliers in deflagration detection.

3. ALGORITHM

3.1 Overview

The multi-camera-system used consists of N cameras, whose camera parameters (focal lengths $f_{x,n}, f_{y,n}$, optical center $c_{x,n}, c_{y,n}$, translation vector t'_n and rotation matrix R_n) can be found by intrinsic and extrinsic calibration (e.g. Zhang (2000)). They are positioned around the space of interest so that their fields of view overlap. Each camera records with a framerate of 200 Hz, leaving 5 ms for computation for all N images.

The algorithm proposed in this paper can be separated in various steps, of which most depend on the outcome of preceding steps. These dependencies lead to a rather unusual order, that differs from that of previous works:

(1) Deflagration Segmentation
(2) 2D-Localization
(3) 3D-Localization
(4) Volume Estimation
(5) Local Detection
(6) Global Detection

Each time a new image arrives at the computer, all steps are executed in the given order. In case of desynchronized cameras, results of steps (1-2) are saved so that they only have to be calculated for the newly arriving images.

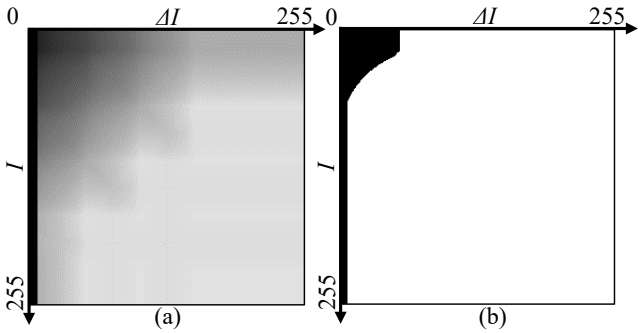


Figure 1. **(a)** Look-up-table LUT_p that describes the membership to the set of deflagration-like pixels. Black pixels have a membership of 0, white pixels a membership of 1. **(b)** Combinations of I and ΔI for which pixels are classified as deflagration-like using a threshold of $\theta = 0.3$.

3.2 Segmentation

The segmentation process uses the two main attributes of a deflagration: Its intensity is rather high and very likely to increase further. For every image i and pixel x, y, the membership to the set of deflagration-like pixels is determined by a heuristic look-up-table LUT_p, based on a fuzzy logic. The two input parameters are the intensity $I_i^{x,y}$ and its increase $\Delta I_i^{x,y} = I_i^{x,y} - I_{i-1}^{x,y}$. Simply said, membership increases with growing $I^{x,y}$ and $\Delta I^{x,y}$ (the exact table used in our experiments is shown in Fig. 1). A threshold θ is then used to separate deflagration-like pixels from non-deflagration-like pixels:

$$D_i^{x,y} = \begin{cases} 1 & LUT_p(I_i^{x,y}, \Delta I_i^{x,y}) \geq \theta \\ 0 & \text{else} \end{cases} \tag{1}$$

This algorithm is rather simple but gives good results compared to its low computational effort. However, without further steps, many misclassified pixels can be expected. Thus, additional criteria have to be met to add a pixel to the detection segmentation map_{det}:

Spectral filtering The spectral attributes of deflagration pixels are checked in RGB and YCbCr space. E.g., the red component r of a combustion can be expected to be higher than the green and blue components g and b:

$$S_i^{RGB} = \begin{cases} 1 & r > b \ \cap \ r > g \ \cap \ r > \overline{r} \\ 0 & \text{else} \end{cases}, \tag{2}$$

where $\overline{r}$ is the mean of the red component over all pixels in the current image, and, slightly modified from Seo et al. (2015),

$$S_i^{YCC} = \begin{cases} 1 & \overline{Y} > 17 \cap Y > \overline{Y} \cap Cb < \overline{Cb} \cap Cr > \overline{Cr} \\ 0 & \text{else} \end{cases}, \tag{3}$$

where Y is the Luminance, Cb the chrominance-blue and Cr the chrominance-red and $\overline{Y}$, $\overline{Cb}$, $\overline{Cr}$ their respective mean over the whole image.

Background-Model Additionally, a background model based on an exponential moving average is implemented.

In the first image, the background B_0 is initialized with the red channel image r_0. The background of each following image is then calculated with

$$B_i = 0.4\ B_{i-1} + 0.6\ r_i. \tag{4}$$

r_i and r_0 are smoothed slightly before calculation. Similar to Jiang et al. (2015), the foreground is then calculated using a variable threshold ΔF, so that it only contains regions of the image in which significant changes take place:

$$F_i = \begin{cases} 1 & |r_i - B_i| > \Delta F^{x,y} \\ 0 & \text{else} \end{cases} \tag{5}$$

with

$$\Delta F^{x,y} = \begin{cases} 12 & r_i^{x,y} > 130 \\ 28 & r_i^{x,y} > 85 \\ 56 & \text{else} \end{cases}. \tag{6}$$

Only if all conditions are met, pixels are added to the detection segmentation:

$$map_i^{det} = D_i \cap S_i^{RGB} \cap S_i^{YCC} \cap F_i. \tag{7}$$

An opening-operation with a small kernel is then applied on map_i^{det}.

To this point, the segmentation algorithm leads to two problems:

- The sensitivity is mainly depending on θ. However, while during detection, high sensitivity is crucial, it also creates way more misclassifications that significantly decrease the quality of 2D- and thus also 3D-localization. A lower sensitivity would lead to increased precision here. Using a compromise with a medium sensitivity decreases quality of both detection and localization.
- The design of the look-up-table and especially the background model exclude pixels whose intensity does not grow further. This is essential to decrease misclassifications (e.g. from light bulbs with a constant intensity) but also often leads to "holes" in the center of a deflagration, where intensity on the sensor reached its maximum despite growing further in reality.

Both problems can be solved by keeping a high sensitivity in the segmentation map^{det}, which is used for detection while introducing another segmentation map^{loc} that is then used for localization. To increase precision, only pixels from map^{det} whose (smoothed) red channel reached its maximum value are transferred to map^{loc} (to increase robustness, n_d preceding detection segmentations are considered here as well). To prevent the occurrence of holes, pixels are kept in map^{loc} as long as the red channel is at its maximum, even if they are no longer present in the detection segmentations:

$$map_i^{loc} = \begin{cases} 1 & (map_{i-1}^{loc} \cup map_i^{det} \cup map_{i-1}^{det} \cup \ldots \cup \\ & map_{i-n_d}^{det}) \cap r = 255 \\ 0 & \text{else} \end{cases}. \tag{8}$$

When used in surroundings with difficult light, a closing operation with a rather big kernel may optionally be used on map^{loc} to increase quality of the volume estimation. In any case, map^{loc} is then transferred to map^{det} to fill the holes there as well:

$$map^{det} \leftarrow map^{det} \cup map^{loc}. \tag{9}$$

3.3 2D-Localization

To locate a deflagration in the image of a camera n, the moment $(x_{2D,n}, y_{2D,n})$ is calculated from map^{loc}. Also, the amount of pixels in both map^{det} ($N_n^{\prime det}$) and map^{loc} ($N_n^{\prime loc}$), as well as their proportions in comparison to image size $N_n^{det} \in [0,1]$ and $N_n^{loc} \in [0,1]$ are calculated, parameters, that are needed later in 3D-localization, volume estimation and detection. To prevent rare cases in which pixels stay in map^{loc} constantly (e.g. by removing a red object in front of a bright background), set pixels in map^{loc} are set to zero in map^{det} and map^{loc} if none of the location segmentations did change its size (so if $N_{n,i}^{loc} = N_{n,i-1}^{loc} \forall n$).

3.4 3D-Localization

From the two-dimensional moments, lines $t_n + \lambda \cdot l_n$ in three-dimensional space may be calculated for every camera n. These start at the respective camera and describe all points on which, according to the camera's view, the deflagration might take place:

$$l_n' = \begin{pmatrix} (x_{2D,n} - c_{x,n})/f_{x,n} \\ (y_{2D,n} - c_{y,n})/f_{y,n} \\ 1 \end{pmatrix}. \tag{10}$$

The resulting lines are transformed into the same coordinate system:

$$t_n = R_n^T \cdot t_n' \tag{11}$$

$$l_n = R_n \cdot l_n'. \tag{12}$$

In a noise- and error-free measurement, lines of detecting cameras will meet at a certain point in space. However, in real measurements, lines will not meet but instead be skew, which makes a simple search for the intersection point impossible.
Thus, these so-called triangulation problems are often formulated as an optimization problem. In this work, for every camera n and every point $p = (x, y, z)^T$ in 3D-space, a cost function ϕ_n is defined, penalizing the closest distance between the line and p quadratically:

$$\phi_n = \frac{\|(p - t_n) \times l_n\|_2^2}{\|l_n\|_2^2} \tag{13}$$

with $\|\cdot\|_2$ being the operator for the euclidean norm. Note that the closest distance is not measured on the (2D-)sensor, which is done in most related literature and penalizes errors depending on the distance between camera and object, but in three-dimensional space. This is done due to the rather big inaccuracies occurring in deflagration localization, which are not related to camera-deflagration distance.

If summed over all cameras with $N^{loc} > 0$ as in

$$\Phi = \sum_{\forall n: N_n^{loc} > 0} \phi_n. \tag{14}$$

the overall cost function Φ will have a single minimum at the point where the sum of squared closest line-point-distances is minimal.
The point p, for which Φ is minimal can then be found by gradient descent or related methods, or analytically by solving the equation system

$$\frac{\partial \Phi}{\partial x} = 0, \frac{\partial \Phi}{\partial y} = 0, \frac{\partial \Phi}{\partial z} = 0 \tag{15}$$

for x, y and z.

The example images in Fig. 2 (g-i) demonstrates that even a few misclassified pixels (due to a reflection, occlusion or other sources of light) may significantly reduce the quality of a position estimate from Eq. 13. This effect may be greatly reduced by assigning a weight to every camera, which is given as the proportion of detected pixels in map^{loc} (more precisely as N_n^{loc}):

$$\phi_n = N_n^{loc} \cdot \frac{\|(p - t_n) \times l_n\|_2^2}{\|l_n\|_2^2} \tag{16}$$

For example, in case of two detecting cameras, the minimum of Φ does not lie halfway on the line that intersects the two points on the two camera lines that are closest to each other but is shifted towards the camera line with the higher weight assigned.

Since N_n^{loc} depends on the distance between camera and deflagration quadratically (a camera close by will generally detect more deflagration-like pixels), deflagrations that are close to a camera will have a very high weight compared to the other cameras. One may attempt to make up for this by adapting the cost function to

$$\phi_n = \|p - t_n\|_2^2 \cdot N_n^{loc} \cdot \frac{\|(p - t_n) \times l_n\|_2^2}{\|l_n\|_2^2}, \tag{17}$$

where cameras are now weighted with the true, distance-independent, perceived size of the deflagration. This cost function may contain multiple minima (for example when camera positioning is poor) which is why the minimum of Φ can not be calculated analytically anymore. However, the minimum of Eq. 17 can be approximated by determining the minimum of Eq. 16 in a first step and then considering the resulting distance information for one or more additional calculations by adapting the weight w *before* calculation:

$$w_{n,t+1} = w_{n,t} \cdot \|p_t - t_n\|_2^2 \tag{18}$$

with $w_0 = N_n^{loc}$. The effect of Eq. 17 and Eq. 18 on localization is situational and does not necessarily increase overall estimate quality. Also, the algorithm might (unlikely) still iterate to different minima, which is why the equalization of the distance is considered a purely optional step.

3.5 Volume Estimation

The exact volume of a deflagration is hard to determine since its shape is irregular and boundaries are often ambiguous. In this work, deflagration volume is roughly estimated based on the assumption that the deflagration is sphere-shaped. This assumption is handy since the appearance of a sphere is the same from every direction and the resulting algorithm is very light-weight, also because results from previous steps can be used during calculation.

First, the radius of the deflagration-sphere is calculated individually for each camera, using the camera parameters and the camera-deflagration-distance to calculate the "surface" $surf_n$ that is linked to the localization segmentation. Note that pixel surface again depends, due to inverse square law, on camera-deflagration-distance quadratically and that the localization segmentation is used as a base due to its increased precision compared to the detection segmentation:

$$surf_n = \frac{1}{f_{x,n} \cdot f_{y,n}} \cdot N_n^{'loc} \cdot \|p_t - t_n\|_2^2. \tag{19}$$

By assuming that $surf_n$ is linked to a circle due to the sphere-model of the deflagration, its radius rad_n can be calculated as in

$$rad_n = \sqrt{\frac{surf_n}{\pi}}, \tag{20}$$

resulting in a radius for every camera. Under idealized conditions, if the deflagration is indeed a sphere that can be seen completely by all cameras, these radii should be identical. In reality, however, mainly due to occlusions but also irregular shapes of deflagrations, the radii will differ from each other. Simply averaging the radii would in most cases result in an underestimation as soon as occlusions occur. Thus, the radii are weighted by themselves, assigning to the camera that sees most of the deflagration the highest weight:

$$Rad = \frac{\sum^N rad_n^2}{\sum^N rad_n}. \tag{21}$$

Alternatively, radius can simply be determined as the maximum radius:

$$Rad = max_n(rad_1, ..., rad_N). \tag{22}$$

From Rad, the volume V of the deflagration is determined:

$$V = \frac{4}{3}\pi \cdot Rad^3. \tag{23}$$

3.6 Local Detection

The classification process of every single camera is considered as local detection. The detection uses, very similar to pixel-wise classification in Eq. 1, a main attribute of a deflagration: It is growing not only in intensity but also in size. Again, a heuristic lookup table LUT_d, based on a fuzzy logic, determines the membership of an image to the class of deflagration images. Input variables are the proportion of detected pixels of the whole image $N_{n,i}^{det}$ and its increase $\Delta N_{n,i}^{det} = (N_{n,i}^{det} - N_{n,i-2}^{det})/\tau$ with τ beeing the difference in time between measurement of images i and $i-2$. A two-frame distance is evaluated to increase stability:

$$defl_i = \begin{cases} 1 & LUT_d(N_{n,i}^{det}, \Delta N_{n,i}^{det}) \geq \theta_d \\ 0 & \text{else} \end{cases} \tag{24}$$

Note that the detection segmentation is used here instead of the localization segmentation for increased sensitivity and early detection. Sensitivity of Eq. 24 can again be controlled by a threshold θ_d but is also depending quadratically on the distance between camera and deflagrations; e.g. a camera close to a deflagration will classify far more pixels to be deflagration-like than a camera far away. To compensate for this, $N_{n,i}^{det}$ (and subsequently $\Delta N_{n,i}^{det}$) is adapted by

$$N_{n,i}^{det,ad} = N_{n,i}^{det} \cdot \frac{\|p_t - t_n\|_2^2}{stdDist^2} \tag{25}$$

before use in Eq. 24. The standard distance $stdDist^2$ is the distance for which LUT_d is designed. If distance is not known (e.g. only one camera's localization segmentation contains any pixels), $\|p_t - t_n\|_2$ is set to $stdDist$, effectively removing the distance compensation. However, this is only relevant when maximum sensitivity is desired (see Section 3.7.

Additional conditions secure a low false-alarm-rate:

Exponential-Based Regression Coefficient The expected, exponential growth rate of a deflagration is checked by

$$EBRC_{n,i} = \frac{N^{det}_{n,i} - N^{det}_{n,i-1}}{3 \cdot (e^2 - e^1)} + \frac{N^{det}_{n,i} - N^{det}_{n,i-2}}{3 \cdot (e^2 - e^0)} + \frac{N^{det}_{n,i-1} - N^{det}_{n,i-2}}{3 \cdot (e^1 - e^0)}. \quad (26)$$

Foreground Alteration Additionally, the number of Foreground pixels $N^F_{n,i}$, its alteration $\dot{N}^F_{n,i} = N^F_{n,i} - N^F_{n,i-1}$ as well as the alterations alteration $\ddot{N}^F_{n,i} = \dot{N}^F_{n,i} - \dot{N}^F_{n,i-1}$ are considered.

For increased robustness, the respective camera's localization segmentation has to be non-empty. Local detection is then done with

$$isdefl_{n,i} = \begin{cases} 1 & (defl_{n,i} \vee defl_{n,i-1} \vee defl_{n,i-2} \vee \\ & defl_{n,i-3}) \wedge EBRC \geq 2.5 \cdot 10^{-5} \wedge \\ & \dot{N}^F_{n,i} > 0 \wedge \ddot{N}^F_{n,i} > 0 \wedge N^{'loc}_n > 0 \\ 0 & else \end{cases}. \quad (27)$$

3.7 Global Detection

For maximum sensitivity, global detection could be done by simply triggering as soon as one camera detects. However, we propose to increase robustness against false alarms by adding as an additional condition that at least two cameras' localization segmentations contain pixels. This also ensures that the deflagration has been localized and distance-compensation in Eq. 25 has been performed.

4. RESULTS

The performance of the proposed algorithm has been tested with videos of real deflagrations. Due to the high effort linked to the generation of deflagrations in sufficient scale, only 15 datasets containing a deflagration are available. All of them contain the image sequences of three cameras, eight datasets with a framerate of 200 Hz and seven of them with 100 Hz. The camera setup in the test cabin is shown in Fig. 3. To check for erroneous detections, 19 videos with possible detection-scenarios containing moving red, white and orange objects like balls, foils and an orange reflective vest, and light sources like flashlights, laser-pointers, lighters and matches have been tested as well (with a comparable camera setup). For all datasets, the same parameters have been used.

Segmentation and 2D-Localization In Fig. 2, some example images of a deflagration at three points in time are shown, including their segmentations map^{det} and map^{loc} as well as their moments, and the backprojected, estimated 3D-locations resulting from Eq. 13, 16, and 17. T_1 is the point of time of the earliest global detection, $T_2 = T_1 + 15$ ms and $T_3 = T_2 + 4$ ms.

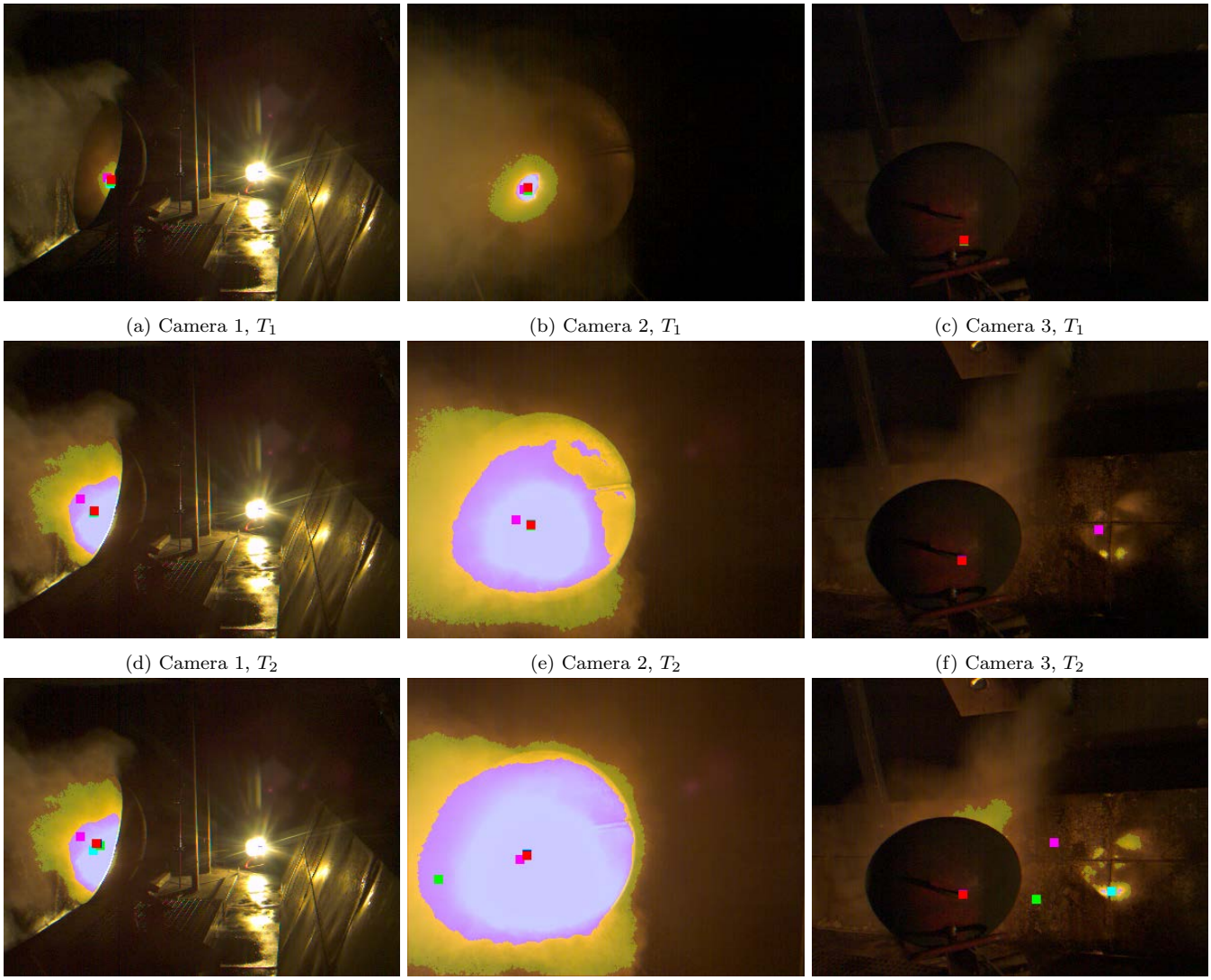
Due to the irregular shape of a deflagration and the absence of a sound ground truth information it is impossible to measure the precision of 2D-localization (as well 3D-localization or volume estimation) with a metric. However, it is obvious that the moments resulting from map^{loc} give a significantly better estimate of the 2D-position of a deflagration than that of map^{det} for the given example. In all other datasets, the perceived precision of the moment of map^{loc} is either (in many cases significantly) better or comparable to the moment of map^{det}.

3D-Localization In the example given in Fig. 2, at T_1 and T_2, only map^{loc} of cameras 1 and 2 contains pixels. Since both segmentations represent the deflagration well, the use of an unweighted localization algorithm as in Eq. 13 results in a reasonable estimate of the deflagrations location. At T_3, map^{loc} of camera 3 contains a few erroneously classified pixels resulting from a reflection way off the actual deflagration. Despite the small number of pixels, the 3D-Localization with Eq. 13 is significantly affected negatively by the respective moment. Weighting the moments according to Eq. 16 greatly increases quality of the location estimate. This effect can be observed for nearly every situation in the data sets where occlusions, reflections or other sources of erroneously classified pixels occur. For every dataset, position estimates of Eq. 16 are either (in many cases significantly) better or comparable to those of Eq. 13.

The distance compensation in Eq. 17 or 18 shows a noticeable effect only if a deflagration reaches its end-phase, e.g. when N^{loc} grows big but does not necessarily give a good estimate of measurement reliability since the deflagration is too big to be seen by the camera completely. However, results are highly situational in such cases and hard to evaluate without ground truth. Overall, we perceive a slight, but rather insignificant increase in localization quality compared to Eq. 16.

Volume Estimation Fig. 4 shows the course of the estimated radii over time for the example dataset shown in Fig. 2 and 3. At the event of the first local detection, volume from maximum radius is estimated as 1.44 l. At T_3, radius is estimated as 126.5 mm (8.49 l) from mean, 209 mm (38.26 l) from weighted mean and 259.1 mm (72.89 l) from maximum. It is important to understand here that not only the volume of the white "core" but also the bright red area around it is estimated. A rough comparison with an object of known size in the test cabin, the parabolic bowl (radius approx. 250 mm) indicates that volume is generally underestimated when using unweighted or weighted mean. When comparing the radii, differences are rather small. However, when comparing volume, deviations may reach significant extent due to the cubic relationship between radius and volume. Using the maximum radius for calculation of the volume, estimation seems to be quite accurate.
In any case, when deflagrations grow too big to be completely seen by a camera, volume estimates are not reliable anymore.

Local and Global Detection During local detection, all deflagration datasets have been classified correctly. From the datasets without deflagration, 5 (26.3%) have erroneously been classified positive. Two of these datasets do either contain a match or a lighter that is ignited directly



(a) Camera 1, T_1 (b) Camera 2, T_1 (c) Camera 3, T_1

(d) Camera 1, T_2 (e) Camera 2, T_2 (f) Camera 3, T_2

(g) Camera 1, T_3 (h) Camera 2, T_3 (i) Camera 3, T_3

Figure 2. Images of cameras 1-3 at points in time T_1, $T_2 = T_1 + 15$ ms and $T_3 = T_2 + 4$ ms of an example dataset. map^{det} is shown as yellow-green overlay, map^{loc} as blue overlay. map^{loc} is always a subset of map^{det}. The moments of each map^{det} are shown as magenta, those of map^{loc} as torqoise squares, the backprojected points from 3D-localization according to Eqs. 13, 16, and 17 as green, blue, and red squares (blue squares mostly hidden behind red squares).

in front of a camera, the third a laser-pointer pointing directly into the camera, the fourth is containing a reflective vest that moves in front of a window to bright sunlight and the last dataset contains an orange warning light. In all cases except the warning light, the other cameras' localization segmentations do not contain any pixels, so that during global detection, only the dataset containing the warning light has not been classified correctly.

For every dataset containing a deflagration, two ground truth starting points of the deflagration have been determined by hand. GT_1 is determined as the point in time when a bright core of the deflagration (not only sparks or illuminated surroundings) is visible in the image of at least one camera for the human eye. GT_2 is determined as the point in time, where the bright core can be seen in the images of at least two cameras. Even with the setup used in our experiments (there is always an occlusion due to the parabolic bowl), the latter will still be linked to a very early phase of the deflagration (for example in Fig. 2 (a-c) T_1 is equal to GT_2).

Fig. 5 shows the temporal offset between GT_1 and the time of first local detection (in our datasets always camera 2) in blue. Only in two datasets the deflagration is detected one image delayed. The difference between GT_1 and global detection is shown in orange. In most cases, global detection is multiple images delayed.

Fig. 6 shows the temporal offset between GT_2 and the time of first local detection in blue. It can be seen that in most datasets, the deflagration is detected locally multiple images before it can be seen by more than one camera. The difference between GT_2 and global detection is shown in orange. In six datasets, the deflagration is detected globally one image delayed.

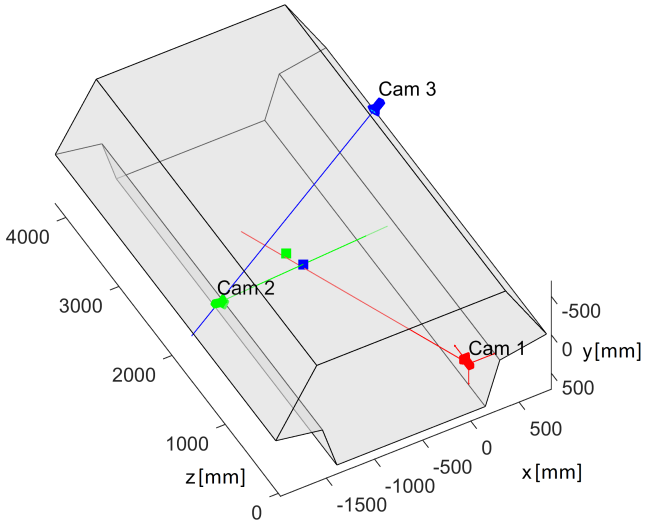


Figure 3. Setup of the test cabin and camera-lines for the example dataset shown in Fig. 2 at T_3. The result of the unweighted localization is shown as green square, that of weighted localization as blue square.

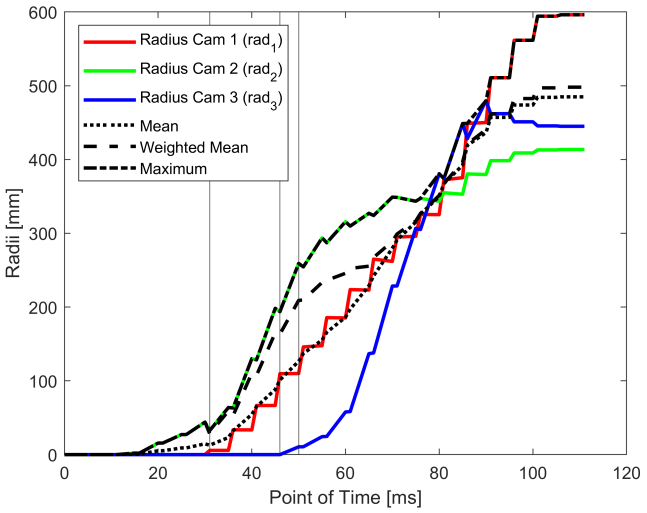


Figure 4. Course of the estimated radius of the dataset shown in Figs. 2 and 3. Vertical lines correspond to T_1, T_2 and T_3. The step-like appearance is a result of asynchrone gathered images.

5. DISCUSSION

Segmentation and 2D-Localization The use of two separate segmentations for detection and localization enables significantly increased precision during localization and volume estimation without losing the high sensitivity needed for early detection; The moments of map^{loc} describe the center of the deflagration way better than that of map^{det}. The interdepending nature of the segmentations also ensures that they always show coherent information. Additionally, the check for pixels in map^{loc} increases resistance to false alarms significantly.

3D-Localization The 3D-Localization greatly benefits from the increased precision of the 2D-estimates. While giving good results in regular cases, an unweighted localization fails in many cases when misclassifications due

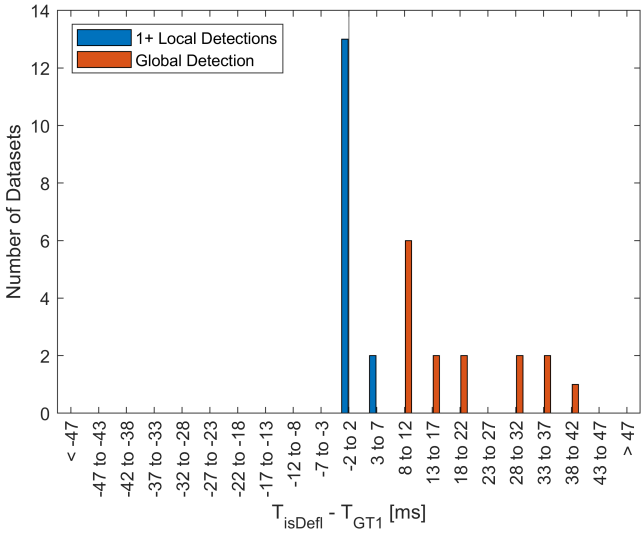


Figure 5. Temporal offset between GT_1 and the first single camera (e.g. local) detection (blue), respectively the global detection (yellow).

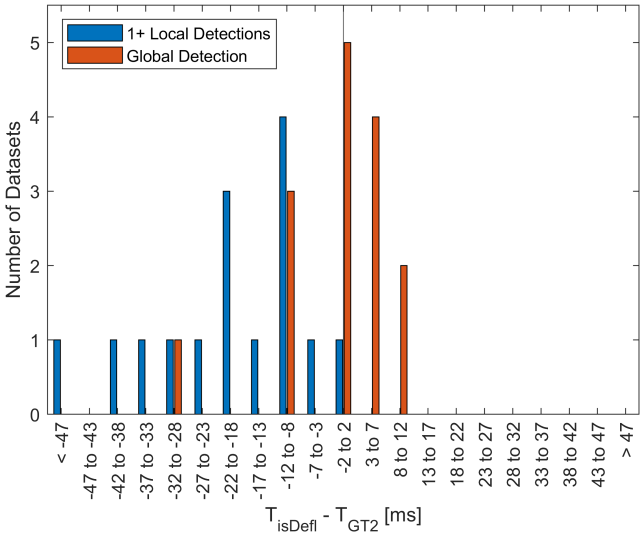


Figure 6. Temporal offset between GT_2 and the first single camera (e.g. local) detection (blue), respectively the global detection (yellow).

to reflections or other sources occur. Dealing with these misclassifications/outliers in the 2D-images is a problem of high complexity and linked to a comparatively high computational effort. The use of a weighted cost function for 3D-localization on the other hand greatly increases robustness in such cases without increasing computational effort noticeably. If the moments used for localization contain no outliers, estimates of weighted and unweighted localization are nearly identical. Distance equalization in Eq. 17 has a situational effect on localization. Relevant differences occur only when the deflagration is in a very advanced state. Whether it yields an overall increase of localization precision is hard to evaluate with our current data set and will be subject of further research.

For our special use case, a cost function that penalizes errors in 3D-space is superior to the current state of the art (penalty on sensor, thus dependent on camera-object-

distance): A very large proportion of the measurement errors is completely independent of camera-object distance (occlusions, reflections, ...). Its use has also the side effect that the minimum can be calculated analytically and thus very efficiently. The latter is of course not true for the distance equalization but use of Eq. 18 gives a good approximation.

Volume Estimation The proposed algorithm for volume estimation is based on the localization segmentations map^{loc} that are calculated anyway and can thus be performed with negligible computational effort. While the assumption of a sphere-shaped deflagration is not accurate, it still enables a rough estimate of volume, which can be used as an additional parameter for extinction.

Due to the occlusions that are present in every positive dataset (due to the parabolic bowl), the proposed algorithm is consistently underestimating the radii of some cameras which affects their mean as well. If the volume is calculated from the mean radius, it is underestimated significantly. A weighted mean of radii as in Eq. 21 mitigates this effect but cannot completely prevent it. A simple consideration of the maximum radius as in Eq. 22 seems to yield quite accurate estimates but may more easily overestimate volume in some situations. However, this has not been observed in our datasets.

Local and Global Detection While single cameras detect deflagrations in a very early stage in most datasets, it is also much more prone to false alarms, mainly when light sources get very close to the sensor. Global detection is delayed by multiple images in many cases but still detects all deflagrations at a very early stage and generates only one false positive classification in our datasets. One should also consider that the dataset used in our experiments contains only deflagrations that are occluded for all but one camera during its initial phase. In a potential real-world scenario, a developing deflagration is likely to be seen immediately by two or more cameras, thus not resulting in a detection delay. The type of detection used should depend on the specific application. E.g. in applications with varying threat of deflagrations, switching between local and global detection may be used to run a "low-sensitivity-mode" with low risk of false alarms, respectively a "high-sensitivity/early-detection-mode" with increased risk of false alarms.

6. OUTLOOK

To allow for a better evaluation of localization and volume estimation, simulations, as well as real experiments with objects of known shape, are planned. These should clarify whether use of Eq., 17, respectively 18, yields an increase in localization quality worth the additional computational effort and the risk of multiple minima in the cost function, and which volume estimation is more reliable (weighted mean, maximum).

Additionally, while most of the algorithm has already been implemented in CUDA and proved to be fast enough, the newest additions to the algorithm have only been tested with MATLAB/c++ and might slightly exceed the maximum runtime. Currently, an FPGA-implementation is investigated. First results promise a significant reduction of runtime as well as hard real-time capability.

To increase robustness regarding light sources like warning lights, the integration of an infrared sensor as additional detection condition is planned.

REFERENCES

Cetin, A.E.e.a. (2013). Video fire detection - review. *Digital Signal Processing*, 23(6), 1827–1843. doi: 10.1016/j.dsp.2013.07.003.

Ernst, T., Kümmerlen, F., and Fay, A. (2016). Real-time image processing based deflagration detection and localization. *Future Security*, 11.

Hartley, R. and Schaffalitzky, F. (2004). l_∞ minimization in geometric reconstruction problems. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 1, 504–509. doi:10.1109/CVPR.2004.1315073.

Hartley, R.I. and Sturm, P. (1997). Triangulation. *Computer vision and image understanding*, 68(2), 146–157. doi:10.1006/cviu.1997.0547.

Jiang, B., Lu, Y., Li, X., and Lin, L. (2015). Towards a solid solution of real-time fire and flame detection. *Multimedia Tools and Applications*, 74(3), 689–705. doi: 10.1007/s11042-014-2106-z.

Kim, A., Liu, Z., and Crampton, G. (2004). Explosion suppression of an armored vehicle crew compartment. *Proceedings of 2004 International Symposium on Safety Science and Technology*, 4, 1070–1074.

Kümmerlen, F., Schröder, T., and Krüger, K. (2013). Image processing based deflagration detection within crew compartments of armoured vehicles. *Proceedings of the 13th International Conference on Fire Science and Engineering*, 265–276. doi:10.1016/j.firesaf.2014.02.004.

Muhammad, K., Ahmad, J., Mehmood, I., Rho, S., and Baik, S.W. (2018). Convolutional neural networks based fire detection in surveillance videos. *IEEE Access*, 6, 18174–18183. doi:10.1109/ACCESS.2018.2812835.

Qi, S., Du, Y., Zhang, P., Liang, J., Wang, S., and Li, Y. (2018). Study on gasoline-air mixture deflagration flame with different equivalence ratios in a closed vessel. *Combustion Science and Technology*, 190(1), 20–31. doi: 10.1080/00102202.2017.1358169.

Schröder, T., Krüger, K., and Kümmerlen, F. (2013). Fast detection of deflagrations using image processing. *Proceedings of the suppression, detection and signaling research and applications symposium.*

Schröder, T., Krüger, K., and Kümmerlen, F. (2014). Image processing based deflagration detection using fuzzy logic classification. *Fire Safety Journal*, 65, 1–10. doi:10.1016/j.firesaf.2014.02.004.

Seo, J., Kang, M., Kim, C.H., and Kim, J.M. (2015). An optimal many-core model-based supercomputing for accelerating video-equipped fire detection. *The Journal of Supercomputing*, 71(6), 2275–2308. doi:10.1007/s11227-015-1382-3.

VDI 2263 (2017). VDI-guideline 2263: Dust Fires and dust explosions: Hazards - assessment - protective Measures.

Zhang, Z. (2000). A flexible new technique for camera calibration. *IEEE Transactions on pattern analysis and machine intelligence*, 22. doi:10.1109/34.888718.