


Particle Tracking and Detection Software for Firebrands Characterization in Wildland Fires

Alexander Filkov* , School of Ecosystem and Forest Sciences, University of Melbourne, 4 Water St, Creswick, VIC 3363, Australia

Sergey Prohanov, Mechanics and Mathematics Faculty, National Research Tomsk State University, 36 Lenin Ave., Tomsk, Russia 634050

Received: 3 April 2018/Accepted: 7 December 2018

Abstract. Detection and analysis of the objects in a frame or a sequence of frames (video) can be used to solve a number of problems in various fields, including the field of fire behaviour and risk. A quantitative understanding of the short distance spotting dynamics, namely the firebrand density distribution within a distance from the fire front and how distinct fires coalesce in a highly turbulent environment, is still lacking. To address this, a custom software was developed in order to detect the location and the number of flying firebrands in a thermal image then determine the temperature and sizes of each firebrand. The software consists of two modules, the detector and the tracker. The detector determines the location of the firebrands in the frame, and the tracker compares the firebrand in different frames and determines the identification number of each firebrand. Comparison of the calculated results with the data obtained by the independent experts and experimental data showed that the maximum relative error does not exceed 12% for the low and medium number of firebrands in the frame (less than 30) and software agrees well with experimental observations for firebrands $> 20 \times 10^{-5}$ m. It was found that fireline intensity below $12,590 \text{ kW m}^{-1}$ does not change significantly 2D firebrand flux for firebrands bigger than 20×10^{-5} m, while occasional crowning can increase the firebrand flux in several times. The developed software allowed us to analyse the thermograms obtained during the field experiments and to measure the velocities, sizes and temperatures of the firebrands. It will help to better understand of how the firebrands can ignite the surrounding fuel beds and could be an important tool in investigating fire propagation in communities.

Keywords: Wildland and structural firebrands, Firebrand detection, Firebrand tracking

1. Introduction

Detection and analysis of the objects in a frame or a sequence of frames (video) can be used to solve a number of problems in various fields, including the field of fire behaviour and risk [1, 2]. During wildland and structural fires, a large number of burning and glowing firebrands are generated and then transported by the convection column of a fire front and by the wind [3]. These firebrands consist of pie-

* Correspondence should be addressed to: Alexander Filkov, E-mail: alexander.filkov@unimelb.edu.au



ces of structural materials, bark and twigs, and can initiate new ignition of fuel ahead of the fire front.

The size distribution of firebrands generated during the combustion of vegetation and structures is relatively unknown in field conditions [4–6]. A quantitative understanding of short distance spotting dynamics, namely the firebrand density distribution with a distance from the fire front and how distinct fires coalesce in a highly turbulent environment, is still lacking [7]. An understanding of how the firebrands can ignite the surrounding fuels is an important consideration in mitigating fire propagation in communities [8]. Therefore, there is a need in the study of ignition and glowing of firebrands and the estimation of the energetic characteristics.

Firebrands can be in flaming and/or smouldering combustion modes, thereby imposing restrictions on recording instruments (ability to determine combustion mode, firebrand's size in the flaming mode, separation “hot” and “cold” firebrands), particularly the limitation to use video cameras. For instance, high-speed videography (420 fps) requires counting the number of glowing firebrands and determining their sizes with reasonable accuracy [9]. The sole remaining way for solving such problems is to use a thermal imager. However, in processing thermal video, specific problems arise that are related to the detection and tracking of firebrands:

1. Firebrands vary in temperature over time, making them difficult to track through thermal characteristics alone;
2. Firebrands can vary in size and shape over time, making them difficult to track through shape metrics alone;
3. Firebrands can be travelling in very different directions from each other, making it difficult to apply a single filter to a video stream to better determine the direction a firebrand is going, and better predict where it ends up later in time;
4. Firebrands can be travelling at a variety of different velocities, making it more difficult to know where a single firebrand maybe in the next frame of the video;
5. Firebrands occur in a landscape that is constantly varying in temperature, making impossible to use detection methods based on background temperature.

With respect to the problems stated above, the existing object tracking approaches [10–55], such as those used in traffic management, facial recognition systems, digital cameras, etc., are not appropriate for thermal video processing from wild-land fires. Usually, algorithms for detecting and tracking the objects are used for the video analysis. Tracking is defined as the determination of the trajectory for the moving objects in the sequence of frames. The tracker can independently solve the problem concerning the detection of the regions containing the object and identify the matches in different frames using the iterative processing of the data obtained in the previous frame. However, there is also a mode of operation in which the location of the objects is determined by the specific detection algorithms.

Table 1 shows a summary of the literature analysis in respect to different methods used for detection and tracking of objects on the video. It gives a description

Table 1
Summary of Detection and Tracking Methods

Methods	Algorithms	Description	Advantages	Disadvantages
1. Detection 1.1 Identification of moving objects in a sequence of frames [10]	1.1.1 Inter-frame difference [11, 22, 33]	Comparison of objects in a sequence of frames	Most simply implemented	In our case, the two inter-frames have significant differences: the background colour, and also the shape, colour and size of the firebrand are changed
	1.1.2 Subtraction of the background [44, 51, 52] (e.g. the Gaussian mixture model [12, 13, 53–55])	The algorithms are based on the subtraction of the background. At each point in the image, the algorithm determines the parameters of Gaussian mixtures separately for the background and the moving objects	Algorithms allow characterize the variability of the background with high accuracy	These algorithms only operate correctly when presented with a slowly changing background and they also require the collection of temporal statistics
	1.1.3 Calculation of the optical flow [14–16]	The optical flow approach allows to determine the displacement of each pixel in two inter-frames in the sequence	High robustness at different lighting conditions	There is a need to meet the conditions such as the constant intensity of each point that belongs to the object in time. All the neighbouring points belonging to the same object have to move at the same velocity
	1.1.4 Mixture of the 1.1.1–1.1.3 approaches	The mixed approaches can combine any options for the movement detection in the frame.	Contain advantages of the algorithms mentioned above.	Contain disadvantages of the algorithms mentioned above.

Table 1
continued

Methods	Algorithms	Description	Advantages	Disadvantages
1.2. Analysis of each frame and solving the problem of classification	1.2.1 Particle's shape or colour analysis (e.g. Support Vector Machine [17–19], the Viola-Jones [20, 21, 23] and convolutional neural networks [24–26] algorithms)	All the test objects can be found in the image by analysing each frame and solving the problem of classification.	The algorithms have high accuracy and do not depend on the change in the temperature (colour) of the background and firebrands The shape or colour of the object allows to identify the test object by analysing only one frame	Most of them require the creation of a database with the different images of objects and all the possible false objects (similar to the firebrands, such as the pieces of background or fingers of fires)
2. Tracking				
2.1 Point tracking	2.1.1 Deterministic algorithms [28, 29]	Deterministic algorithms are characterized by the verification of matches between all the objects in the previous frame and each object in the current frame	Easy to solve and high processing speed. High accuracy, only two frames are enough for the analysis. Can be used for the quick-changing video stream and for the large number of firebrands in the frame	The matching error function is usually defined using specific metrics, such as the proximity of the objects between the frames, the same direction of movement, the external similarity, etc.
	2.1.2 Probabilistic algorithms (the Kalman filter [31, 32, 34] and the particle filter [35, 36])	Probabilistic algorithms are based on the concept of the state space [30] and suggest that the moving object has a particular state that is changed in each frame. To determine the state of the object in the new frame, all the previous states of the object are analysed in the previous frames	They use information about moving object, such as velocity and direction	They can be used without modification, only for singular objects. These methods require the use of additional algorithms

Table 1
continued

Methods	Algorithms	Description	Advantages	Disadvantages
2.2 Kernel tracking	2.2.1 Mean-shift algorithms [41–43]	The objects are tracked by determining the motion of the kernel in the sequence of frames (kernel is commonly called the shape or appearance of the object)	Fast processing, because they do not perform a large number of calculations. They do not require the form of the object	These algorithms have a high efficiency only when there is a significant difference between the color of the object and background
2.3. Silhouette based	2.3.1 Shape matching [45–47] and counter tracking [48, 49] algorithms	For each frame, the tracker of the silhouette detects the location of the object using the model that comprises the previous states of the object (in the previous frame)	They allow to work with the objects of a complicated shape that cannot be described by a simple geometric figure	Objects of unspecified shapes cannot be detected

of methods found, as well as advantages and disadvantages for firebrand identification.

The literature review [10–49, 51–55] has shown that currently there are no off-the-shelf approaches that provide a solution to the stated problem. To detect flaming and glowing firebrands, the optimally suitable methods are the methods based on a comparison of the shape with a certain pattern (Table 1, method 1.2). The point-tracking method (Table 1, algorithm 2.1.1) can be used to compare the objects in different frames. These methods do not depend on the change in the temperature of the (colour) background and firebrands (problems 1 and 5) and can be used for the quickly changing video stream, as well as for the large number of firebrands in the frame (problems 2–4). The other advantage of this approach is that it does not require the time-consuming compilation of a database for all types of firebrands.

2. Methods

In our work we used firebrand production data obtained by El Houssami et al. [4], Filkov et al. [56] and Thomas et al. [57] as a result of the experimental fires conducted as prescribed fires. The forest type in the experiments [4, 56, 57] was pitch-pine scrub-oak, dominated in the canopy by pitch pine, with intermittent clusters of post-oak and white oak in the sub-canopy. The understory contained a shrub layer of huckleberry, blueberry, and scrub oaks. More details related to the experiment can be found in [4, 56–58].

In these experiments, the thermal imager FLIR A325 was used to film firebrands. The thermal imager operated in the range of 7.5–13 microns with frame resolution 320×240 pixels and frequency of 8 Hz. To record the firebrands in the air flow, the screen was installed in front of the thermal imager [56]. The screen was a 2.5×1.5 m gypsum wall board installed perpendicular to the underlying surface and coated with black heat-resistant paint. In front of the screen 20 aluminium trays, 0.3×0.24 m each were positioned to form a rectangle with a total cross section area of 1.4 m^2 . The thermal imager was focused on the objects located 0.5 m from the screen (focal plane). It was assumed that 0.5 m in both directions from focal plane is acceptable for detection and tracking of firebrands. As a result, our working domain was $1.5(\text{W}) \times 1.5(\text{H}) \times 1(\text{D})$ m. More details related to the equipment arrangement can be found in [56].

Data from thermal imager was obtained in SEQ format (the radiometric sequence file of the imager FLIR A325) and converted to MATLAB format using the FLIR ThermaCAM Researcher software. The received files contained the number of the test frame, the recording time in milliseconds and the matrix with a temperature at each point of space.

The IR video file processing task was to detect the location of flying firebrands, determine the temperatures and sizes and calculate the number of the firebrands that fell to the surface under study. The most difficult part of the data processing was to build a firebrand tracker to detect the new location of the firebrand in the next frame. When 30 firebrands and more (Fig. 1) appear in the frame, it is diffi-

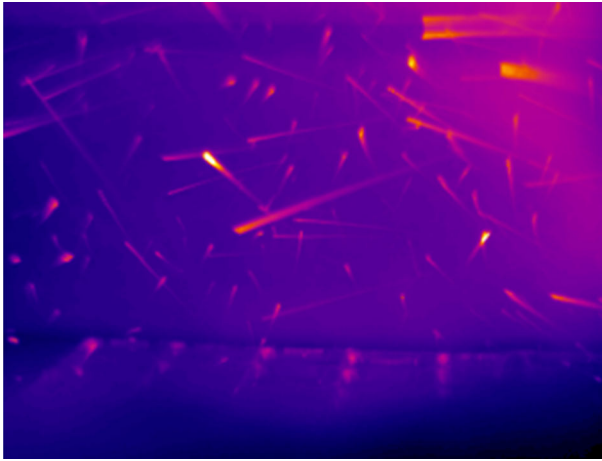


Figure 1. Snapshot with firebrands captured by the thermal imager.

cult to determine a new location of a certain firebrand in the next frame. This is due many firebrands having similar sizes.

The review of the works [10–49, 51–55] has shown that there is no complete solution for determining the characteristics of moving firebrands. In this connection, to solve this problem, we used a computer vision library OpenCV [59]. The problem is divided into two sub-tasks: the task for determining the location of the firebrands in the frame (detector) and the task for comparing the firebrands in different frames and determining the identification number of each firebrand (tracker) [60].

2.1. Detector

A special detector was built to detect the firebrand location. The operation of the detector is to process each frame and provide the coordinates of the rectangular area that contains a firebrand. Despite data obtained by the thermal imager allowing us to determine the temperature at each point of the frame, sometimes it was very difficult to separate the background from the moving firebrand. The colour representation of the background and firebrands temperatures varies significantly between frames; however, in the same frame they differ by only on a few degrees (Fig. 2). It is impossible to select a single threshold temperature that would allow us to distinguish particle from background. A custom designed algorithm was developed to solve this problem.

The idea of the algorithm is to select the special filters which allow us to separate the area by increasing contrast that contains a track of the flying firebrand. In general, the flying firebrand creates a linear track (Fig. 1 and 2), which is an artefact caused by the low frame frequency (8 Hz) of the video. The thermal imager cannot record a firebrand at a specific point due to the high velocity of the firebrand, resulting in a diffuse view of the firebrand.

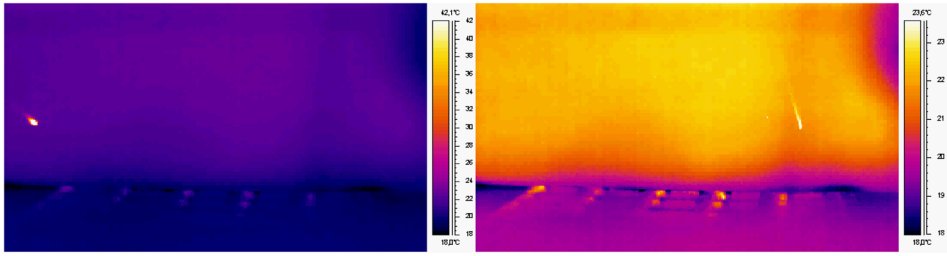


Figure 2. Frames with a firebrand at different moments of time. The minimum temperature is 290 K, the maximum temperature is 300 K in the frame.

Since the firebrand can fly in any direction, there is a need for a set of filters which will increase the response for the specifically-directed firebrand track. To build the filters, it is reasonable to use the function which most closely reproduces the track of the firebrand.

In this technique, each filter is a two-dimensional matrix of coefficients, the values of which can be regarded as the values of a discrete function $f(i,j)$ that depends on two parameters. In this case, the filters are cylindrical surfaces (Gaussian functions elongated along the specified direction):

$$f(i,j) = \exp\left(-\frac{(-i * \sin(\theta) - j * \cos(\theta))^2}{\sigma^2}\right), \quad (1)$$

here θ is the angle along which the Gaussian is elongated, σ is the Gaussian function parameter, $i, j = -\frac{n}{2}, \frac{n}{2}$ are matrix indexes and n is a filter size (empirically chosen).

The size of the filter matrix, the number of directions and the parameter σ are empirically chosen (using fitting approach) in such a way as to gain a maximum response in the areas where the firebrands on filtered frames are located. The accuracy was evaluated for ≥ 20 frames and ≥ 10 firebrands on each frame.

The image processing by the convolution filtering is as follows:

$$I'(x,y) = I(x,y) * f = \sum_{i=1}^n \sum_{j=1}^n I\left(x - \frac{n}{2} + i, y - \frac{n}{2} + j\right) \cdot f(i,j), \quad (2)$$

where f is the filter, n is the filter size, $I(x,y)$ is the temperature of the original image at the point (x,y) , $I'(x,y)$ is the image convolution with a filter at a point (x,y) .

The convolution filtering method provides the greatest response if the centres and directions of the filter and the firebrand track are the same. After the convolution of the frame with all filters we obtain the filtered frames, the highest values of which correspond to the firebrand tracks. Using the built-in functions of the

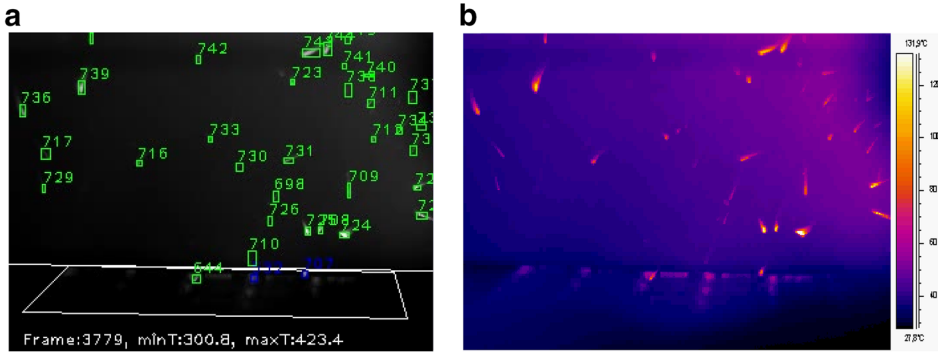


Figure 3. Operation of the detector (a) for the given frame (b).

computer vision library (OpenCV [59]), we build a rectangular region around each group of pixels.

Then the firebrands are selected by size. If the firebrand size is defined as insignificant (less than 10^{-5} m^2) or is very large (more than $1.7 \cdot 10^{-3} \text{ m}^2$, the firebrand is not in focus), the firebrand is not detected. The minimum and maximum firebrand size was selected in accordance with [4]. To determine the temperature of the firebrand, the maximum temperature was found in the frame limited by a detection rectangle.

The use of this approach allows us to detect most of the firebrands and determine the required parameters (Video 1 in Electronic supplementary material). The detector steadily operates for the entire video (Fig. 3). In addition to receiving the coordinates of the firebrand, the direction of movement is also determined, which is required for the operation of the tracker.

2.2. Tracker

After the detection of all firebrands in the frame, the analysis is conducted to determine whether these firebrands are new or also located in the previous frame and assign a new identification number to the firebrands in accordance with this data. To implement this approach, we developed a firebrand tracker. The development of the tracker was greatly complicated by the following reasons:

1. High velocity of firebrands. The firebrand can potentially fly across the screen in 1–2 frames.
2. The large number of firebrands in some frames (Fig. 1). At the same moment, two firebrands might only be slightly different from each other, in either size or direction of movement.

To compare the detection of the firebrands in two sequential frames, four special metrics are calculated for each pair (d_i, \tilde{a}_j) . Here d_i is the detection of the firebrand in the current frame, \tilde{a}_j is the detection in the previous frame, where i

and j are the numbers of the detections on the current and previous frames respectively.

2.2.1. Similarity of the Firebrands The first metric shows the similarity of the detections (d_i, \tilde{d}_j) . The detection-pair regions are bounded to the same size (the biggest detection size) across the frames. Then the detection-pair regions are transformed into a binary format, i.e. the original image is converted into a black-and-white image, the pixels of which have only two values 0 and 1. The background in the detection region is excluded by the value of the threshold temperature. The threshold temperature is automatically determined for each detection. It was calculated using the following equation: $T_{th} = T_{min} + (T_{max} - T_{min}) * 0.4$, where T_{min} and T_{max} are the minimum and maximum temperatures in the detection region respectively, 0.4 is the empirical parameter. After that we determined which of the pixels have equal temperatures. The more similar the detections are, the closer the value of the metric to 1. Each detection in a previous frame is compared with all detections in the current frame.

2.2.2. Similarity of the Directions This metric indicates whether the detection d_i is in the current frame in the required location with respect to the detection \tilde{d}_j in the previous frame. It takes a value between 0 and 1. The possibility of the location is checked by the direction and velocity of the firebrand. The analysis of the thermal video showed that there were some chaotically moving firebrands that changed the direction by 90° or more. However, these firebrands were in the minority, and for this reason we excluded firebrands whose trajectory deviated in the next frame by more than 45°. Figure 4 shows a firebrand with a visible track. The black rectangle is the detection region of the firebrand. The distances $d1$ and $d2$ were selected according to the maximum and minimum velocity of the firebrands in the video for the different firebrand sizes.

To calculate this metric, we determine the direction of the firebrand. If the firebrand d_i deviates from this direction by an angle α more than 45°, then the value of the metric is equal to 0. If the deviation of the metric is less than 45°, then the closer the direction of the firebrand d_i to the direction of the firebrand \tilde{d}_j is, the closer the metric to 1.

The third metric takes the value 1 if the detection of the firebrand d_i in the current frame is located at a certain distance from the detection \tilde{d}_j and takes the value 0 if this condition is not met. The detection d_i is assumed to be located at a distance more than $d1$ in the next frame (Fig. 4). This distance depends on the size of the detection. The larger the size of the detection, the greater distance a firebrand can fly for the time between the frames.

The fourth metric describes the proximity of the firebrand detections to a straight line in the last three frames. Analysis of the video showed that most firebrands had linear trajectory, due to short distances between frames. The parameters of the straight line passing as close as possible to the points A, B, C, D, E, F are determined by the least square method (Fig. 5). The sum of the distances is

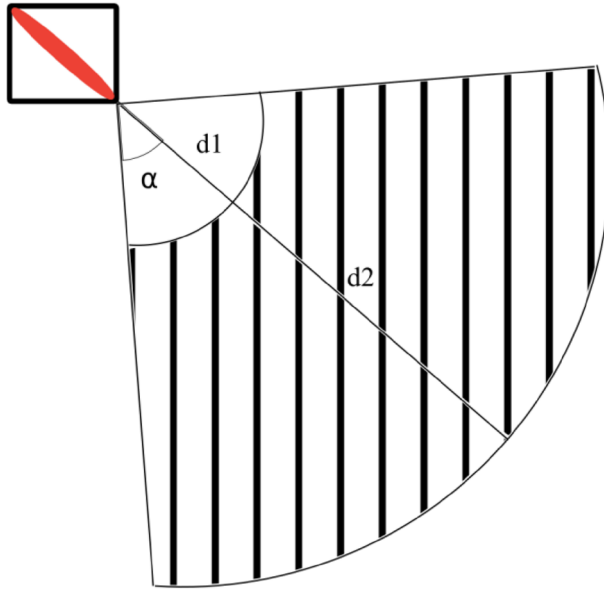


Figure 4. Determination of the similar directions of the firebrands. Here $d1$ and $d2$ are the minimum and maximum distance that the firebrand flies for the time between frames, α is the maximum angle of deviation from the linear trajectory.

calculated from these points to the straight line and is normalized so that in the case when the sum of the distances is 0, then the metric would take the value 1, and when the sum of the distances increases, then the metric would tend to 0.

Finally, the sum of all metrics $m(i, j) = f_1(d_i, \tilde{d}_j) + f_2(d_i, \tilde{d}_j) + f_3(d_i, \tilde{d}_j) + f_4(d_i, \tilde{d}_j)$ is calculated for each pair of the detections in the two sequenced frames. Each row of the matrix contains the maximum value, and if it exceeds an empirically chosen threshold (that was chosen to maximize the accuracy of the tracker operation), then the detection d_i in the current frame belongs to a firebrand \tilde{d}_j in the previous frame, or the detection d_i belongs to a new firebrand.

After processing the data, the software creates the video file that contains all regions of the detections, the identification numbers of the firebrands, the frame number, and the minimum and maximum temperature in the frame (Video 2 in Electronic supplementary material).

To count the number of the firebrands that fell on the unit of surface area per unit of time, the rectangular region (Fig. 3a) was built in each frame of the thermal video. The number of the firebrands in the required area is calculated using the following procedure. The principle of this procedure is to extract the last detection (coordinates of the rectangle delimiting the detection) in the track from the database for each identification number of the firebrands and then to check whether the detection is located in the required area. If the detection is in the database, then the flag InTray is set to the position TRUE.

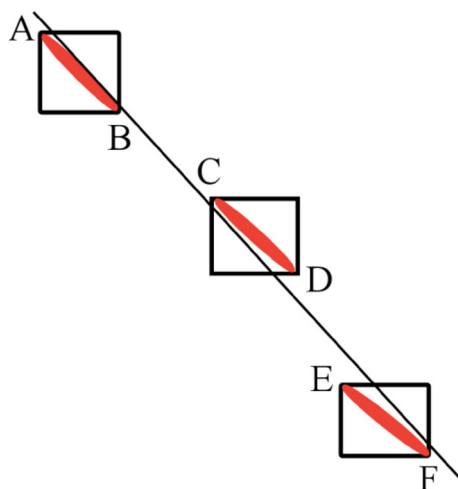


Figure 5. Determining the proximity of the firebrand trajectories to a straight line.

To determine the firebrand size, the binary matrix is extracted from the database, where 0 corresponds to the background within the rectangle of the detection, and 1 corresponds to the firebrand. Then, the firebrand size is checked, and if the firebrand length is more than 20 cm, the firebrand is ignored, since it was assumed to be located outside the focus area of the thermal imager. Analysis of the literature [4–6, 57] showed that firebrands longer than 20 cm have not been recorded during wildfires and we assumed that this value can be used as a threshold. The sum of the matrix content (i.e. area) is calculated using a standard library function (*numpy*) for the firebrands selected. The firebrand area is transformed to our metric system by multiplying the firebrand area by a factor. To determine this factor we used information about the size of the screen, trays [56] and thermal imager resolution. Since the firebrand area can vary from frame to frame, the average, maximum and minimum firebrand area is calculated for the entire length of the track.

The information on the firebrand position is extracted from the database to determine the velocity of the firebrand, i.e. the coordinates for the rectangles of the detections in each frame belonging to the track of the firebrand. Then the distance travelled by the firebrand is calculated for each firebrand. The average firebrand velocity is defined as the distance travelled by the firebrand, multiplied by the factor and divided by the time, i.e. the travelled track length divided by the number of frames per second. For some firebrands it is not possible to count the velocity, as the firebrand track consists of the single frame and these firebrands cannot be considered.

To determine the average temperature of the firebrand, the values of all temperatures for the detections belonging to a firebrand along the entire track were selected from the database. The values were then averaged.

3. Results and Discussion

The data obtained after the processing of the thermal video were entered in the MongoDB [61] document-oriented database management system (DBMS) that was optimized for storing the hierarchical structures and used the NoSQL-approach. MongoDB was chosen due to the simple design, high performance, fast entry of data in the database, as well as the possibility of ad-hoc queries to extract the required data.

Each entry in the database is a string in the JSON format. It displays a frame number (*frameNumber*), a temperature of the firebrand (*temperature*), a detection region of the firebrand (*detRegion*, an array of values), and the values of all temperatures in the detection region (*content*, an array of values). The flag *inTray* indicated whether the firebrand fell to the area under study and the parameter *id* was a unique identifier of the firebrand.

For the timely processing of data there is a need to maintain a separate collection that indicates the minimum and maximum temperature in each frame. For a visual representation, all information is saved in an XML formatted file that contains the minimum and maximum temperature and the information on the detection of firebrands for each frame. The section “*Detections*” stores the characteristics of each detection region, i.e. the sequence number of the detection, the coordinates of the rectangular area that describes the region of the detection, the temperature of the firebrands, as well as the special flag “*InTray*” that indicates if the firebrand is located in the region.

The software developed allows a user to detect and count the number of firebrands on the video, calculate speed, temperature (mean, max or min) and cross section area of them, and determine a major direction of flying firebrands. Figure 6 presents some examples of such data, which can be obtained using the developed software.

It should be noted that minimum requirements for the developed software are 320x240 frame resolution and 8 Hz IR video frequency. Based on the domain size and the average velocity of firebrands, the frame rate below 8 FPS will reduce the performance of the tracker, as the distances between firebrands in the inter-frames will increase significantly and it will be difficult for the tracker to match the firebrands. In addition, decreasing of the frame rate will lead to errors in determining velocity of firebrands. It also should be mentioned that low temperatures of firebrands (Fig. 6d) are not related to our software, as it used temperature scale from the pre-recorded experimental IR video.

As we had a limited number of IR videos, the software was tested only for velocities of firebrands less than 11 m/s and fire intensities less than $12,590 \text{ kW m}^{-1}$. For higher velocities and fire intensities it requires additional testing. Analysis of the results also showed that accuracy of the software is challenging to evaluate when the number of firebrands become more than 30 in the frame. This is not necessarily a weakness of the software, but one that arises because there is no way to verify the software output with manual counts.

To verify the accuracy of the software, the video was split into three sets of intervals: low number of firebrands in the frame (< 5), medium (5–30) and high

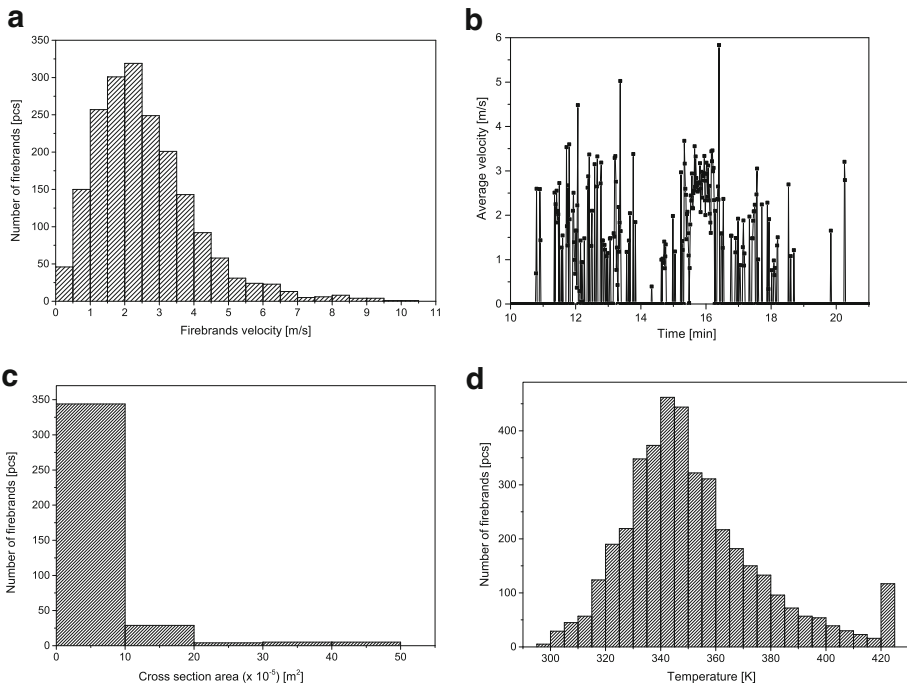


Figure 6. Examples of software usage: (a) velocity of firebrands, (b) average velocity of firebrands per second versus time of the experiment, (c) cross section area of firebrands, (d) mean temperature of firebrands.

(> 30). These intervals were selected for the software to estimate the number of flying firebrands and validate its effectiveness as their number increased. Three independent experts counted the number of flying firebrands not knowing the data the program obtained. The relative measurement errors were calculated according to this data. The average value of the measurement data obtained by the experts was used as the exact value. For the first set of intervals with low numbers of firebrands in the frames a 100% match was obtained. The data calculated for the second set are given in Table 2.

Table 2 shows that the maximum relative error does not exceed 12% for the selected intervals. It is seen that the accuracy of the software decreases from a low to medium number of firebrands. For the third set of intervals the comparison was not made, as the number of the firebrands was very high and it was not possible for experts to track firebrands between frames.

At the next step we compared the results of video processing with observational data of [56, 57] (Fig. 7). Specifically, we compared firebrand flux distribution as a number of firebrands fallen on one square meter of area per second (hereafter 2D firebrand flux). It was not possible to do a quantitative comparison of 3D firebrand flux (distribution of firebrands in 1 m^3 of the air flow), due to the absence

Table 2
Comparison Between the Number of Flying Firebrands q and the Relative Measurement Errors δ for the Medium Number of Firebrands (5–30) in the Frames

Frames	Program data (pcs.)	Expert 1 (pcs.)	δ (%)	Expert 2 (pcs.)	δ (%)	Expert 3 (pcs.)	δ (%)
1800–2250	61	69	12	67	9	65	6
3200–3650	41	46	12	43	5	41	0
4100–4400	31	33	6	32	3	33	6

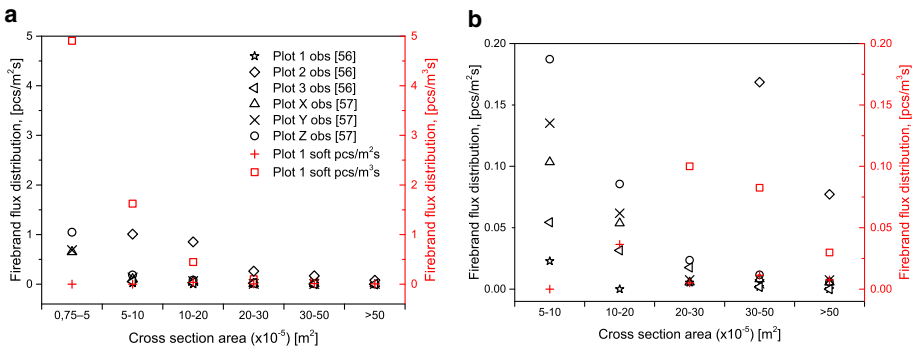


Figure 7. Comparison of firebrand flux size distributions: (a) is a whole range of fluxes, (b) is a magnified region of largest size classes. Black colour represents experimental data of [56, 57] (left Y axis) and red colour results of video processing. Data for 0.75–5 range in [56] are lacking. Both figures have the same legend (Color figure online).

of data. However, we did a qualitative comparison of 3D firebrand flux (Fig. 7). In our processing we used IR video recorded on Plot 1 [56]. Firebrand fluxes in Fig. 7 are calculated as a total number of firebrands divided by the time interval between first and last firebrand (similar approach used in [57]).

Before analysing Fig. 7, it is worthwhile mentioning fireline intensities during experimental surface fires [56, 57]. In [56] fireline intensities were in the range 180 kW m^{-1} to 4370 kW m^{-1} [58], while in [57] intensities ranged between $7350 \pm 3480 \text{ kW m}^{-1}$ (Plot X) to $12,590 \pm 5870 \text{ kW m}^{-1}$ (Plot Z). In [58] was mentioned that in some parts of the parcel crowning was observed with fireline intensity reaching 21,000 kW m^{-1} . Unfortunately in [57, 58] the authors did not provide fireline intensities for each plot. However, even if fireline intensities are known, it is challenging to couple them with firebrand flux. It is known [57], that for such kind of fires firebrand showers occurred up to 100 m ahead of the fire front and therefore, it is very difficult to link specific fire intensity with firebrand

flux. Nevertheless, Thomas et al. [57] found that collection area with highest firebrand flux was correlated to the highest fire intensity and the lowest flux to the lowest fire intensity.

Analysis of 2D firebrand flux [pcs/m²s] for Plot 1 (Fig. 7b) shows that we have a good agreement of our results with observations for firebrands bigger than 20×10^{-5} m. As we expected, the variance for small firebrands ($< 20 \times 10^{-5}$ m) was the most significant. It is not possible to make any qualitative estimations, as we had zero firebrands obtained by software in the range $5\text{--}10 \times 10^{-5}$ m and zero firebrands collected on the Plot 1 in the range $10\text{--}20 \times 10^{-5}$ m. This result could either be related to the limitation of the software, i.e. detection of the firebrands which are not in focus and therefore they are shifting to another range, or limitation of the collection methodology of firebrands during field experiments, i.e. small firebrands could fall apart at landing or blow away. Surprisingly we detected zero firebrands for $0.75\text{--}5 \times 10^{-5}$ m and $5\text{--}10 \times 10^{-5}$ m ranges. However, we detected firebrands below 0.75×10^{-5} m. This result requires more investigation.

Comparing 2D firebrand flux for all experimental and processed data for firebrands bigger than 20×10^{-5} m, we found that they have close values (0.0236 pcs/m²s variation) of flux (Fig. 7b) irrespective to fireline intensities. Except of Plot 2 [56], where crowning was observed and fireline intensity reached $21,000 \text{ kW m}^{-1}$ [58]. For smaller firebrands ($< 20 \times 10^{-5}$ m) we observed 44 times bigger variation (1.0473 pcs/m²s, Fig. 7). We can assume that for considered experimental surface fires the change of fireline intensity ($180\text{--}12,590 \text{ kW m}^{-1}$) has bigger influence on 2D firebrand flux for smaller firebrands ($< 20 \times 10^{-5}$ m). Also, it is clearly seen that even occasional crowning increases firebrand flux significantly (3 to 10 times) compared to surface fire.

Quantitative comparison of 3D firebrand flux [pcs/m³s] showed (Fig. 7a) that it was always higher than 2D [pcs/m²s] for all ranges, which is expected. The only exception was for Plot 2 [56], which could be associated with the isolated regions of crown fires and respectively much higher intensities.

4. Conclusion

The special software was developed to detect the location of flying firebrands, determine the temperatures and sizes, and calculate the number of the firebrands. Comparison of the calculated results with the data obtained by the independent experts and experimental data [56, 57] showed that the maximum relative error does not exceed 12% for the low and medium number of firebrands in the frame (less than 30) and software agrees well with experimental observations for firebrands $> 20 \times 10^{-5}$ m. It was found that fireline intensity below $12,590 \text{ kW m}^{-1}$ does not change significantly 2D firebrand flux for firebrands bigger than 20×10^{-5} m, while occasional crowning can increase the firebrand flux in several times.

Analysis of the thermograms obtained in the field experiments provide a unique opportunity to calculate the number, size, velocity, and thermal characteristics of firebrands generated by wildland fires, and firebrands falling to the fuel bed in the specified volume, per unit of time. These results can support the development of

theoretical models for the firebrand transport during wildland fires in the future. This software has also potential to be used for structural fires, as they produce firebrands similar to wildland fires [57, 62]; it can also be used for analysis of other records, where technology allows to visualize object on the frames.

Future research will be focused on improvement of detection and tracking algorithms, especially for small firebrands using laboratory experiments, as well as using a stereo IR approach to avoid the 2D limitations and measure a 3D distribution of firebrands.

Acknowledgements

This work was supported by the Russian Foundation for Basic Research (Project #18-07-00548), the Tomsk State University Academic D.I. Mendeleev Fund Program and the Bushfire and Natural Hazard Cooperative Research Centre.

ELECTRONIC SUPPLEMENTARY MATERIAL

The online version of this article (<https://doi.org/10.1007/s10694-018-0805-0>) contains supplementary material, which is available to authorized users.

References

1. Shu XM, Yuan HY, Su GF et al (2006) A new method of laser sheet imaging-based fire smoke detection. *J Fire Sci* 24:95–104. <https://doi.org/10.1177/0734904106055568>
2. Magidimisha E, Griffith DJ (2017) Remote optical observations of actively burning biomass fires using potassium line spectral emission. In: *Proceedings of SPIE—the international society for optical engineering*
3. Albin FA (1983) Transport of firebrands by line thermalst. *Combust Sci Technol* 32:277–288. <https://doi.org/10.1080/00102208308923662>
4. El Houssami M, Mueller E, Filkov A et al (2016) Experimental procedures characterising firebrand generation in wildland fires. *Fire Technol* 52:731–751. <https://doi.org/10.1007/s10694-015-0492-z>
5. Gould J, McCaw W, Cheney N et al (2008) *Project vesta: fire in dry eucalypt forest: fuel structure, fuel dynamics and fire behaviour*. CSIRO Publishing, Clayton
6. Manzello SL, Foote EID (2014) Characterizing firebrand exposure from wildland-urban interface (WUI) fires: results from the 2007 Angora fire. *Fire Technol* 50:105–124. <https://doi.org/10.1007/s10694-012-0295-4>
7. Cruz MG, Sullivan AL, Gould JS et al (2012) Anatomy of a catastrophic wildfire: the Black Saturday Kilmore East fire in Victoria, Australia. *Forest Ecol Manag* 284:269–285. <https://doi.org/10.1016/j.foreco.2012.02.035>
8. Pagni PJ (1993) Causes of the 20 October 1991 Oakland Hills conflagration. *Fire Saf J* 21:331–339. [https://doi.org/10.1016/0379-7112\(93\)90020-Q](https://doi.org/10.1016/0379-7112(93)90020-Q)
9. Sharifian A, Hashempour J (2016) A novel ember shower simulator for assessing performance of low porosity screens at high wind speeds against firebrand attacks. *J Fire Sci* 34:335–355. <https://doi.org/10.1177/0734904116655175>

10. Hammami M, Jarraya SK, Ben-Abdallah H (2011) A comparative study of proposed moving object detection methods. *J Next Gener Inf Technol* 2:56–68
11. Cheng YH, Wang J (2014) A motion image detection method based on the inter-frame difference method. *Appl Mech Mater* 490–491:1283–1286
12. Chen Z, Ellis T (2014) A self-adaptive Gaussian mixture model. *Comput Vis Image Underst* 122:35–46
13. Friedman N, Russell S (1997) Image segmentation in video sequences. In: *Proceedings of 13th conference on uncertainty in artificial intelligence*
14. Horn BKP, Schunck BG (1981) Determining optical flow. *Artif Intell* 17:185–203
15. Holte MB, Moeslund TB, Fihl P (2010) View-invariant gesture recognition using 3D optical flow and harmonic motion context. *Comput Vis Image Underst* 114:1353–1361
16. Fortun D, Bouthemy P, Kervrann C (2015) Optical flow modeling and computation: a survey. *Comput Vis Image Underst* 134:1–21
17. Cortes C, Vapnik V (1995) Support-vector networks. *Mach Learn* 20:273–297
18. Heisele B, Ho P, Wu J, Poggio T (2003) Face recognition: component-based versus global approaches. *Comput Vis Image Underst* 91:6–21
19. Ko BC, Cheong K-H, Nam J-Y (2009) Fire detection based on vision sensor and support vector machines. *Fire Saf J* 44:322–329
20. Viola P, Jones M (2001) Rapid object detection using a boosted cascade of simple features. In: *Proceedings of the IEEE computer society conference on computer vision and pattern recognition*, pp 1511–1518
21. Viola P, Jones MJ (2004) Robust real-time face detection. *Int J Comput Vis* 57:137–154
22. Zhang X-B, Liu W-Y, Lu D-W (2008) Automatic video object segmentation algorithm based on spatio-temporal information. *J Optoelectron Laser* 19:384–387
23. Ali I, Dailey MN (2012) Multiple human tracking in high-density crowds. *Image Vis Comput* 30:966–977
24. Ng CW, Ranganath S (2002) Real-time gesture recognition system and application. *Image Vis Comput* 20:993–1007
25. Perlin HA, Lopes HS (2015) Extracting human attributes using a convolutional neural network approach. *Pattern Recognit Lett* 68:250–259
26. Li B, Chellappa R, Zheng Q et al (2001) Experimental evaluation of FLIR ATR approaches—a comparative study. *Comput Vis Image Underst* 84:5–24
27. Yilmaz A, Javed O, Shah M (2006) Object tracking: a survey. *ACM Comput Surv* 38:770–773
28. Veenman CJ, Reinders MJT, Backer E (2001) Resolving motion correspondence for densely moving points. *IEEE Trans Pattern Anal Mach Intell* 23:54–72
29. Shafique K, Shah M (2003) A non-iterative greedy algorithm for multi-frame point correspondence. In: *Proceedings of the IEEE international conference on computer vision*, pp 110–115
30. Kitagawa G (1987) Non-gaussian state—space modeling of nonstationary time series. *J Am Stat Assoc* 82:1032–1041
31. Welch G, Bishop G (2006) An introduction to the Kalman filter. *Practice* 7:1–16
32. Muñoz-Salinas R, Aguirre E, García-Silvente M (2007) People detection and tracking using stereo vision and color. *Image Vis Comput* 25:995–1007
33. Sifakis E, Tziritas G (2001) Moving object localisation using a multi-label fast marching algorithm. *Signal Process Image Commun* 16:963–976
34. Jang D-S, Choi H-I (2000) Active models for tracking moving objects. *Pattern Recognit* 33:1135–1146
35. Nummiaro K, Koller-Meier E, Van Gool L (2003) An adaptive color-based particle filter. *Image Vis Comput* 21:99–110

36. Del Bimbo A, Dini F (2011) Particle filter-based visual tracking with a first order dynamic model and uncertainty adaptation. *Comput Vis Image Underst* 115:771–786
37. Reid DB (1979) An algorithm for tracking multiple targets. *IEEE Trans Autom Control* 24:843–854
38. Blackman SS (2004) Multiple hypothesis tracking for multiple target tracking. *IEEE Aerosp Electron Syst Mag* 19:5–18
39. Tissainayagam P, Suter D (2005) Object tracking in image sequences using point features. *Pattern Recognit* 38:105–113
40. Polat E, Yeasin M, Sharma R (2003) Robust tracking of human body parts for collaborative human computer interaction. *Comput Vis Image Underst* 89:44–69
41. Comaniciu D, Meer P (2002) Mean shift: a robust approach toward feature space analysis. *IEEE Trans Pattern Anal Mach Intell* 24:603–619
42. Bugeau A, Pérez P (2009) Detection and segmentation of moving objects in complex scenes. *Comput Vis Image Underst* 113:459–476
43. Leichter I, Lindenbaum M, Rivlin E (2010) Mean Shift tracking with multiple reference color histograms. *Comput Vis Image Underst* 114:400–408
44. McKenna SJ, Jabri S, Duric Z et al (2000) Tracking groups of people. *Comput Vis Image Underst* 80:42–56
45. Huttenlocher DP, Noh JJ, Rucklidge WJ (1993) Tracking non-rigid objects in complex scenes. In: 1993 IEEE 4th international conference on computer vision, pp 93–101
46. Li B, Chellappa R, Zheng Q, Der SZ (2001) Model-based temporal object verification using video. *IEEE Trans Image Process* 10:897–908
47. Kang J, Cohen I, Medioni G (2004) Object reacquisition using invariant appearance model. In: *Proceedings—international conference on pattern recognition*, pp 759–762
48. Yilmaz A, Li X, Shah M (2004) Contour-based object tracking with occlusion handling in video acquired using mobile cameras. *IEEE Trans Pattern Anal Mach Intell* 26:1531–1536
49. Chen Y, Rui Y, Huang TS (2001) JPDAF based HMM for real-time contour tracking. In: *Proceedings of the IEEE computer society conference on computer vision and pattern recognition*, pp I543–I550
50. Tohidi A, Kaye NB (2017) Comprehensive wind tunnel experiments of lofting and downwind transport of non-combusting rod-like model firebrands during firebrand shower scenarios. *Fire Saf J* 90:95–111. <https://doi.org/10.1016/j.firesaf.2017.04.032>
51. Davis JW, Sharma V (2007) Background-subtraction using contour-based fusion of thermal and visible imagery. *Comput Vis Image Underst* 106:162–182
52. Sobral A, Vacavant A (2014) A comprehensive review of background subtraction algorithms evaluated with synthetic and real videos. *Comput Vis Image Underst* 122:4–21
53. Stauffer C, Grimson WEL (1999) Adaptive background mixture models for real-time tracking. In: *Proceedings of the IEEE computer society conference on computer vision and pattern recognition*, pp 246–252
54. Zivkovic Z, Van Der Heijden F (2006) Efficient adaptive density estimation per image pixel for the task of background subtraction. *Pattern Recognit Lett* 27:773–780
55. Park S, Aggarwal JK (2006) Simultaneous tracking of multiple body parts of interacting persons. *Comput Vis Image Underst* 102:1–21
56. Filkov A, Prohanov S, Mueller E et al (2017) Investigation of firebrand production during prescribed fires conducted in a pine forest. *Proc Combust Inst* 36:3263–3270. <https://doi.org/10.1016/j.proci.2016.06.125>
57. Thomas JC, Mueller EV, Santamaria S et al (2017) Investigation of firebrand generation from an experimental fire: development of a reliable data collection methodology. *Fire Saf J* 91:864–871. <https://doi.org/10.1016/j.firesaf.2017.04.002>

58. Mueller EV, Skowronski N, Clark K et al (2017) Utilization of remote sensing techniques for the quantification of fire behavior in two pine stands. *Fire Saf J* 91:845–854. <https://doi.org/10.1016/j.firesaf.2017.03.076>
59. OpenCv (2014) OpenCV Library. In: OpenCV website
60. Filkov A, Prohanov S (2015) IRaPaD. Automatic detection for the characteristics of moving particles in the thermal infrared video: No. 2015617763
61. MongoDB (2016) MongoDB architecture guide. MongoDB White Paper 1–16
62. Suzuki S, Manzello SL, Lage M, Laing G (2012) Firebrand generation data obtained from a full-scale structure burn. *International Journal of Wildland Fire* 21:961–968. <https://doi.org/10.1071/WF11133>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.