

Adaptive Estimation of Visual Smoke Detection Parameters Based on Spatial Data and Fire Risk Index

Marin Bugarić^{a,*}, Toni Jakovčević^a, Darko Stipaničev^a

^a*Faculty of Electrical Engineering, Mechanical Engineering and Naval Architecture,
University of Split, 21000 Split, Croatia*

Abstract

Standard wildfire smoke detection systems detect fires using remote cameras located at observation posts. Images from the cameras are analyzed using standard computer vision techniques, and human intervention is required only in situations in which the system raises an alarm. The number of alarms depends largely on manually set detection sensitivity parameters. One of the primary drawbacks of this approach is the false alarm rate, which impairs the usability of the system. In this paper, we present a novel approach using GIS and augmented reality to include the spatial and fire risk data of the observed scene. This information is used to improve the reliability of the existing systems through automatic parameter adjustment. For evaluation, three smoke detection methods were improved using this approach and compared to the standard versions. The results demonstrated significant improvement in different smoke detection aspects, including detection range, rate of correct detections and decrease in the false alarm rate.

Keywords: Smoke detection, Wildfire, GIS, Augmented reality, Image

*Corresponding author

Email addresses: marin.bugaric@fesb.hr (Marin Bugarić),
toni.jakovcevic@fesb.hr (Toni Jakovčević), dstip@fesb.hr (Darko Stipaničev)

Preprint submitted to Computer Vision and Image Understanding

1. Introduction

Throughout history, we have witnessed the devastating power of wildfires. Once a wildfire has expanded, it is almost impossible to control and extremely difficult to extinguish. Early identification of wildfires and a quick reaction are the primary factors required to prevent material damage and save human lives.

Experience has indicated that in most cases of wildfires, smoke is visible long before the flame. This phenomenon is particularly evident in environments with dense vegetation. In forests, a flame is not visible until it catches the crowns, meaning that the wildfire is already expanding rapidly. Detecting smoke is therefore one of the most widespread approaches to wildfire detection.

Traditional smoke detection is based on observers who monitor the surrounding environment. Observation posts, located in elevated areas with good visibility, provide visual coverage of most of the surrounding terrain. Advances in technology have led to camera-based surveillance, allowing the observer to control multiple observation posts from a single remote location. This approach has reduced the number of required observers. However, the human factor is still a main problem in remote surveillance. The concentration required to simultaneously monitor several locations is significant, and human attention can decrease over time.

Automatic fire detection systems have been developed to further reduce the number of people and the amount of effort required to provide reliable

smoke detection [1–8]. These systems use different computer vision algorithms, consisting of several detection phases. Most of the systems use motion detection as the first phase of the detection process to reduce the amount of data for analysis. The detected regions are then further analyzed based on visual features, such as color, texture and shape. In another of the general detection phases, the dynamic behavior is analyzed to reliably discriminate between smoke regions and visually similar phenomena. Based on these analyses, the system determines whether to employ the alarm in the given situation.

However, the reliability of detection systems is still not adequate for highly critical applications, such as the smoke detection. It is important to emphasize that smoke detection systems are not fully automatic and rather considered to be an aid to the observers. Human intervention is required only in situations where the system raises an alarm.

The number of alarms depends largely on the detection sensitivity. Most of the existing smoke detection methods provide detection parameters that can be manually edited to change the smoke detection sensitivity. High sensitivity may result in a large number of false alarms, while a low sensitivity may result in failing to spot a fire.

To reduce the number of false alarms and simultaneously increase the number of correct detections, we propose the automatic adjustment of the smoke detection parameters. Our solution is based on integrating computer vision techniques with advanced augmented reality and GIS technologies. We have developed a novel system based on augmented reality used to calculate geographic coordinates and real-world distances of each pixel in the

input image. A geographic information system is then used to modify these data into appropriate information used to adaptively estimate the appropriate sensitivity parameters for the smoke detection. Although some existing systems use different types of synthetic data to improve detection quality [9], the main advantage of our system is the ability to calculate and apply synthetic data, such as the real-world dimensions of visible objects in the image and fire risk indexes, on a pixel level with high accuracy. We also propose modifying the input image by using the calculated data to achieve higher-quality smoke detection.

Note that our intention was not to develop a new smoke detection method. Rather, we propose improving smoke detection in general, by integrating several methodologies. These methodologies work together to provide reliable smoke detection, characterized by a smaller number of false alarms and extended detection range.

A detailed study of existing smoke detection methods was carried out to investigate the most common phases in smoke detection methods. We thoroughly studied each phase and presented ideas for improvements (in Sect. 2). In Sect. 3, we describe the development of the augmented reality system used to calculate the geographic coordinates and real-world distances of each pixel in the input image. The data preparation using GIS and the actual adaptive estimation of the detection parameters are described in Sect. 4 and 5, respectively. We evaluated the proposed improvement using video sequences gathered in the Mediterranean area of Croatia. Several methods were improved by the proposed model and compared to the standard versions (in Sect. 6).

2. Overview of Visual Smoke Detection

There are many approaches to detecting smoke using computer vision. Most of the smoke detection systems simultaneously use several approaches to improve performance and reliability. Several phases of smoke detection are common to most smoke detection systems; therefore, we will cover them in the following subsections. We will also provide ideas and propose improvements based on the integration of computer vision, augmented reality and GIS.

2.1. Motion Detection

Motion detection is a common phase in most smoke detection systems. This phase is used as a filter for subsequent, computationally demanding, phases such that only the detected regions are used for further analysis. There are many motion detection approaches used in smoke detection systems, such as adaptive background estimation [2, 10–13], block mean difference [14–16] or motion history image [17, 18]. One parameter is common to all these approaches: the sensitivity of the motion detection.

Motion detection sensitivity determines the ratio between the reliability and usability of the system. High motion detection sensitivity may result in a large number of false alarms, while low sensitivity may result in a missed detection. For most of the existing methods, motion detection sensitivity is applied to the entire image, as one global parameter.

An important factor in determining the level of sensitivity is the dynamics of the observed location. The overall dynamics of the image are low when the camera is located far away from the region of interest, and increases inversely

proportional to the distance. This is because the local motion of the objects in the scene (e.g., tree branches moving in the wind) is accentuated when these objects are close to the camera.

As mentioned in the previous section, we propose a system that can estimate the geographic coordinates and real-world distances of the observed objects in the image. These data can then be used for the automatic adaptation of motion detection sensitivity. These parameters do not have a single value, but rather different values for different parts of the image. More specifically, a distant movement requires high motion sensitivity, while a movement in the immediate vicinity of the camera requires low sensitivity.

2.2. Region Analysis

Candidate region analysis is another phase common to the majority of smoke detection systems. Regions identified during the motion detection phase are further analyzed based on their color, texture, shape and size.

Smoke is usually light to dark gray, so most of the standard methods use certain chromatic limitations. There are different approaches to chromatic analysis [19–21] using various color spaces. Additionally, some systems use the texture of the candidate region to improve the reliability of the system [22–24]. Different texture features could be obtained by using various approaches, such as a gray-level co-occurrence matrix (GLCM) [25] or wavelet analysis [26–28]. Another aspect of region analysis is examining the shape of the region. The authors in [2] show that smoke regions generally have a convex contour and an irregular shape with many aberrant lines. Most of the methods calculate the shape disorder parameter for the detected region and compare it to the reference values [19, 29], or use it as an input parameter

for a neural network [30].

Methods based on the color, texture and shape properties cannot be significantly improved by using augmented reality and GIS. However, the perceived size of the detected region is directly dependent on the relative distance from the camera. The majority of the existing methods use a threshold, defined by the number of pixels, as the minimum size of the detected region. In our approach, the geographic location of the observed region is known, so GIS could be used to estimate its real-world size. We introduce a new threshold based on the estimated size expressed in a standard unit of measurement, such as meters.

2.3. Dynamics Analysis

An analysis of the smoke dynamics is used for additional verification of the candidate regions. The regions are tracked during a predefined timeframe to validate consistency of smoke-like behavior. One of the primary features of interest is the growth rate of the detected region [3, 10]. Smoke regions should exhibit gradual growth, as well as upward and lateral motion, especially in the incipient phase of a wildfire. The accurate growth rate of smoke regions is rather difficult to predict, as it is significantly affected by various influences, such as the wind, temperature or terrain configuration.

Existing smoke detection methods define the growth rate of the detected regions by the number of pixels, similar to the region size parameter. A smoke phenomenon located a short distance from the camera, appears larger in the input image than a phenomenon of equal size located at a greater distance. Therefore, expressing the growth rate as the number of pixels could lead to false conclusions. Our solution to this problem is to express the

value of growth rate parameter in standard units of measurement by using the augmented reality and GIS technology. In this way, the growth rate of the smoke region is not dependent on the type of the camera used for the detection or the location of the observed phenomena in the input image.

2.4. Decision Phase

Information about the candidate regions is accumulated throughout the detection process. The features are analyzed in the decision phase to provide the final decision about the specific region. This phase occurs only for the candidate regions that have not been eliminated in one of the previous phases.

The standard methods take several aspects of the candidate regions into account, such as the amount of chromatic deviation from the reference values, the texture deviation based on texture descriptors, the dynamics of the region and other aspects specific to different detection systems. There are different approaches to the decision process, such as the Bayesian approach [12], neural networks [10, 22], random forests [31–33], support vector machines [34], the mechanism of thought [4], and others.

We propose introducing a new parameter in the decision phase. This parameter represents a local sensitivity level based on a fire risk index for the observed region. The geographic information system calculates the risk index based on the vegetation type, terrain configuration, meteorological conditions and other factors (explained in detail in Sect. 4.2). This parameter indicates how strong the region features need to be trigger an alarm. This approach allows the system sensitivity to dynamically adapt to specific conditions and situations in the environment where the detection takes place.

3. Scene Calibration Based on Augmented Reality

The first step towards adaptively estimating the visual smoke detection parameters is calculating of spatial data, more specifically, geographic coordinates and physical distances, for each pixel in the input image. This process is called scene calibration, where the goal is to find a link between the real world and the digital elevation model based on precise topography. This model is henceforth referred to as the virtual terrain. This calibration can be achieved using the augmented reality methodology, or, in other words, using a combination of computer vision and computer graphics techniques. Note that the primary goal of our scene calibration process is not to display the virtual terrain, but to use the information gathered from the position and the orientation of the camera with respect to the virtual terrain to calculate the spatial data for each pixel in the input image.

We have developed a new system dedicated to this process. The spatial data have to be further processed to achieve the proposed improvement, so an interface to GIS has been developed. More detail about GIS and calculation of these data is described in Sect. 4.

In the following subsections we will describe in more detail the methods we used to achieve the desired results.

3.1. Scene Calibration and Registration Errors

In recent years, real-time terrain visualization has become an integral part of many GIS technologies [35–37]. Although GIS applications offer a wide range of advanced terrain manipulation functionalities, they are not designed as augmented reality systems and cannot be used during the scene

calibration process. Therefore, it is necessary to develop a scene calibration system with an interface to GIS.

While it was possible to develop such a system from scratch, we chose to use Capaware as a basis for our system [38]. This is open source software that, among other functionalities, has the ability to visualize a virtual terrain overlaid with other 3D models. Only developing a plugin for this software was not feasible, as we needed to modify and improve the core of the Capaware software to achieve a fully functional augmented reality and terrain visualization system.

For scene calibration purposes, each camera from the real world is represented by a virtual camera. Augmented reality systems combine physical objects with computer-generated images. For this reason, it is crucial to find the most suitable model of a virtual camera that best corresponds to a real camera. A properly chosen virtual camera model ensures a reduction in the number of registration errors.

Finding a suitable model for the virtual camera is associated with the problem of 3D objects registration inside an augmented reality system. The extrinsic parameters in our solution are based on information from the sensors that define the orientation and location of the real camera. The intrinsic parameters are based on an accurate calibration of the real camera. We calculate the intrinsic parameters of the real camera by using the method proposed in [39]. Because the virtual camera model (OpenGL camera) is not defined by intrinsic and extrinsic parameters, but a completely different set of parameters, we must find equations to convert from one model to another. With the known intrinsic and extrinsic parameters of the real camera, it is

possible to determine parameters of the virtual camera to create identical camera views and achieve accurate alignment of the real and the virtual environment. This is further described in Sect. 3.2.

The virtual terrain is a digital elevation model based on precise topography. Therefore, each point on this terrain is correctly georeferenced and has a corresponding point in the real world. A method of representing the surface of the real world on a plane must be chosen for our application. An experimental analysis was carried out, and several map projections were considered for our virtual terrain model. We propose the projection defined by EPSG:900913, a close variant of the Mercator projection. The experimental results are out of scope of this paper, considering the quality and widespread usage of this projection.

The spatial data can be extracted from the virtual terrain, and each vertex point can be expressed by the specific coordinates that correspond to the accurate geographic coordinates in the real world. For each pixel inside a virtual camera frame, a ray can be cast and the corresponding coordinates of the virtual terrain visible at those specific pixel coordinates can be found. Finally, if the scene calibration is executed successfully, the real and the virtual cameras share the same view, and for each pixel in the image of the real camera, it is possible to calculate the correct geographic coordinates of the point in the real world.

Cameras used for surveillance and monitoring usually cover an 360° area and in this way control the entire region of interest. Surveillance and monitoring can either be conducted manually by the firefighting operators, or, more commonly automatically using up to 64 predefined camera positions

and zoom levels (preset positions). Because the majority, if not all smoke detection algorithms, operate in automatic mode most of the time and detect smoke only when the camera is stationary, it is reasonable to compute the required parameters only when the camera is positioned at particular preset position. In this way, we can eliminate the registration errors caused by computational delay and unexpected camera movement. The GIS data can be prepared in advance for each preset position, and thus the chosen smoke detection algorithm can be run in real-time without compromising the speed of the original algorithm.

As mentioned, a connection between computer vision and computer graphics is used to develop a scene calibration system based on augmented reality. To our knowledge, only a few works compare the image formation pipelines in computer vision and computer graphics [40]. Some other works try to find spatial information without this connection [41] for the purpose of obtaining the precise position of a forest fire point. Here we describe the methods that we use to calculate the spatial data for the scene visible in the camera input image.

3.2. Connection Between Image Formation Pipelines in Computer Vision and Graphics

For the representation and definition of the real camera we use a pinhole camera model, defined by intrinsic and extrinsic parameters. Eq. (1)-(2) show the transformations and parameters required for the calculation of the pixel coordinates in the image plane of the real camera.

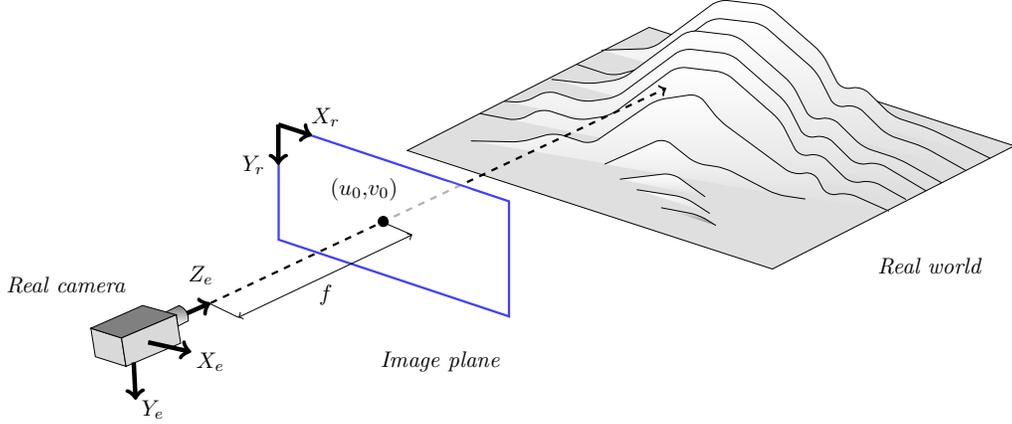


Figure 1: Image formation in the image plane of a real camera

$$x(X_r, Y_r) = P X_w \quad (1)$$

$$P = K[R|T] = \begin{bmatrix} f_x & 0 & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix} [R|T] \quad (2)$$

where (X_r, Y_r) are the Cartesian coordinates of the point x in the real camera image with the origin in the upper-left corner of the image plane, as shown in Fig. 1. Point X_w is a point in the real world that is visible as the point x in the real camera image. The coordinates of the point X_w are defined in the three-dimensional Cartesian coordinate system used by the EPSG:900913, map projection with the origin on the equator and the x-axis along the equator, where the third coordinate corresponds to the height of the terrain. The matrix P is the projection matrix, the matrix K represents the intrinsic parameters of the real camera, where f_x, f_y are the focal lengths

in pixels, and u_0, v_0 define the optical center. The extrinsic parameters are defined by the matrix R , representing the rotation matrix and T , representing the translation vector.

Fig. 1 shows the graphical representation of the image formed by the real camera with all the required parameters. The focal length f in the Fig. 1 is measured in a unit of length (such as millimeters) and should be converted to pixels. Because the pixels are not always square, the focal length should be measured in both directions (f_x, f_y) .

The location and the orientation of the camera define the translation vector T and the rotation matrix R , respectively. If this information is known, any point in the real world (X_w) can be expressed as the eye coordinates (X_e, Y_e, Z_e) , with the camera as the origin of the eye coordinate system. Therefore, the coordinates (X_r, Y_r) of the point x in the image plane of the real camera can be calculated using Eq. (3)-(4).

$$X_r = f_x \frac{X_e}{Z_e} + u_0 \quad (3)$$

$$Y_r = f_y \frac{Y_e}{Z_e} + v_0 \quad (4)$$

On the other hand, we use an OpenGL camera as the representation of the virtual camera. Fig. 2 shows the graphical representation of the image formation pipeline in OpenGL, and the general coordinate transformations occurring in the image formation process of a virtual camera.

Every point on the virtual terrain is georeferenced using the same Cartesian coordinate system used by the EPSG:900913 map projection. The coordinates of these points are transformed into the eye coordinates X_e, Y_e, Z_e

using the modelview matrix. Thus, the modelview matrix corresponds to the extrinsic parameters of a real camera. For this case, we use the perspective projection as seen in Fig. 2(a). The next step is using a projection matrix to transform the eye coordinates to clip coordinates X_c, Y_c, Z_c , as shown in Fig. 2(b). The clip coordinates are transformed into normalized device coordinates $X_{NDC}, Y_{NDC}, Z_{NDC}$ using the perspective division. The normalized device coordinates of all the visible objects range from -1 to 1 for all three axes, as seen in Fig. 2(c). Finally, the normalized device coordinates are scaled and transformed into the window coordinates using the viewport transformations (Fig. 2(d)).

The projection matrix is defined by the field of view angle (in degrees) in the y direction ($fovy$), the aspect ratio that determines the field of view in the x direction ($aspect$), the distance from the viewer to the near clipping plane ($zNear$) and the distance from the viewer to the far clipping plane ($zFar$). The projection matrix is shown in Eq. (5).

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \\ W_c \end{bmatrix} = \begin{bmatrix} \frac{\cot(fovy/2)}{aspect} & 0 & 0 & 0 \\ 0 & \cot(fovy/2) & 0 & 0 \\ 0 & 0 & \frac{zFar+zNear}{zNear-zFar} & \frac{2*zFar*zNear}{zNear-zFar} \\ 0 & 0 & -1 & 0 \end{bmatrix} \begin{bmatrix} X_e \\ Y_e \\ Z_e \\ 1 \end{bmatrix} \quad (5)$$

Eq. (6) shows the transformation process from the clip coordinates directly into the window coordinates. In this equation x_0 and y_0 specify the lower left corner of the viewport rectangle, in pixels, while $width$ and $height$ specify the width and height of the viewport, as shown in Fig. 2(d). Param-

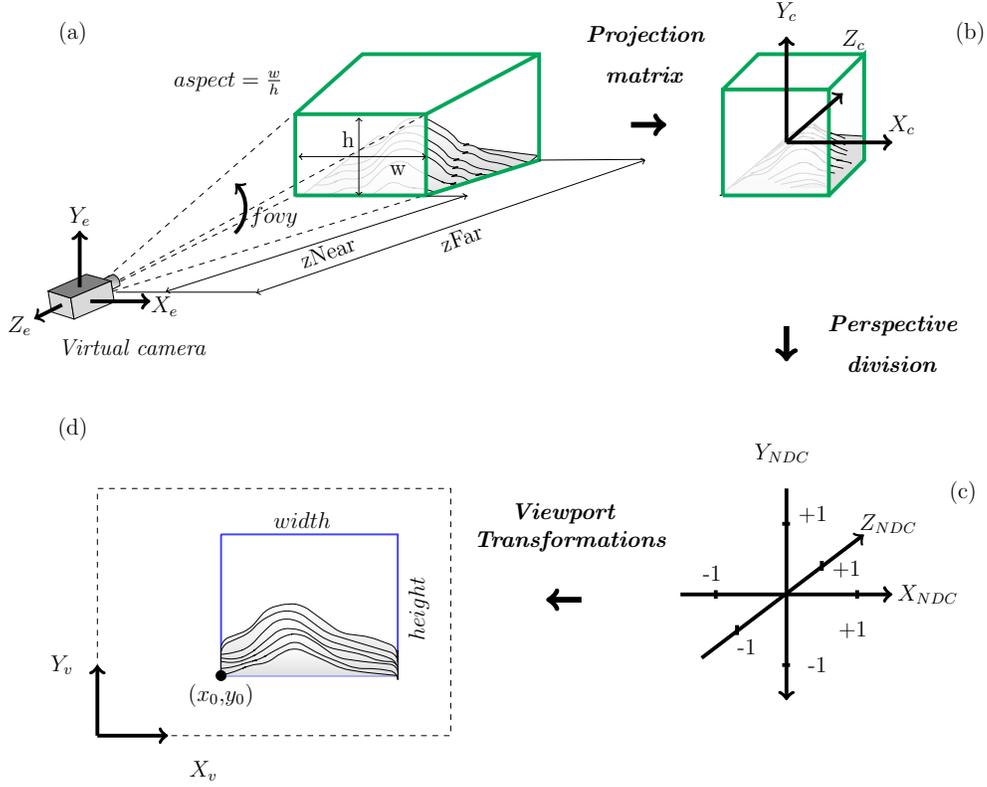


Figure 2: Image formation in the image plane of a virtual camera

eter s is the scaling factor of the homogeneous pixel coordinate.

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{width}{2} & 0 & \frac{width}{2} + x_0 \\ 0 & \frac{height}{2} & \frac{height}{2} + y_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} \quad (6)$$

Figs. 1 and 2 show that the coordinate systems of the virtual and real cameras do not share the same orientation. Therefore, Eq. (7)-(8) are used to calculate the coordinates on the screen of the virtual camera (X_v, Y_v) .

$$X_v = \frac{\cot(fovy/2)}{aspect} \left(\frac{X_e}{Z_e} \right) \frac{width}{2} + \left(\frac{width}{2} + x_0 \right) \quad (7)$$

$$Y_v = -\cot(fovy/2) \left(\frac{Y_e}{Z_e} \right) \frac{height}{2} + \left(\frac{height}{2} + y_0 \right) \quad (8)$$

Because the modelview matrix corresponds to the extrinsic parameters of the real camera, here we inspect the differences between the intrinsic parameters of the real camera and the OpenGL parameters that define the view of the virtual camera. We can assume that (X_r, Y_r) from the Eq. (3)-(4), and (Y_v, Y_v) from the Eq. (7)-(8) represent the same point on the screen of the same dimensions $(width, height)$. However, there is a difference in the orientation of the pixel coordinate systems in OpenGL and the real image. If this is taken into consideration, we can calculate the following Eq. (9)-(12) that describe the conversion process from the intrinsic parameters that define the real camera to the OpenGL parameters that define the virtual camera and vice versa.

$$x_0 = u_0 - \frac{width}{2}, \quad u_0 = x_0 + \frac{width}{2} \quad (9)$$

$$y_0 = \frac{height}{2} - v_0, \quad v_0 = \frac{height}{2} - y_0 \quad (10)$$

$$fovy = 2 * \cot^{-1}\left(\frac{2f_y}{height}\right), \quad f_y = \frac{height}{2 * \tan\left(\frac{fovy}{2}\right)} \quad (11)$$

$$aspect = \frac{width}{height} \frac{f_y}{f_x}, \quad f_x = \frac{width}{height} \frac{f_y}{aspect} \quad (12)$$

Eq. (9)-(12) are the basis for the development of the scene calibration system. The intrinsic parameters of the real camera f_x, f_y, u_0 and v_0 , can now be transformed into the parameters of the virtual camera, $x_0, y_0, fovy$

and *aspect*. Note that these parameters only need to be calculated once for a single camera.

Because the real and the virtual terrain are both georeferenced using the same coordinate system, the application of these equations results in the same view for the virtual and the real camera. Recall that for a successful scene calibration process, the location and orientation of the camera must be known as well. An example of a real and a virtual camera that share the same view is given in Fig. 3.

Information such as the geographic location is stored in the virtual terrain. Also note that the distance from the virtual camera to any point on the virtual terrain is stored in the z-buffer of OpenGL. Therefore, as the final result of the scene calibration process, each pixel in the input image (either real or the virtual) is associated with both a geographic location and a distance from the camera.

4. Pixel Coverage and Fire Risk Index Data Preparation

In the previous sections, we described how to calculate the correct geographic coordinates and relative distances from the camera for each pixel of the input image. These data represent sufficient information to estimate the dimensions of the objects visible in the scene, as well as to estimate the fire risk in a particular part of the image.

To estimate the real-world dimensions of visible objects in the image, we have developed a concept of a pixel coverage map, where each pixel holds the information about the real-world dimensions expressed in standard units of measurement. Further, using GIS and additional information such as

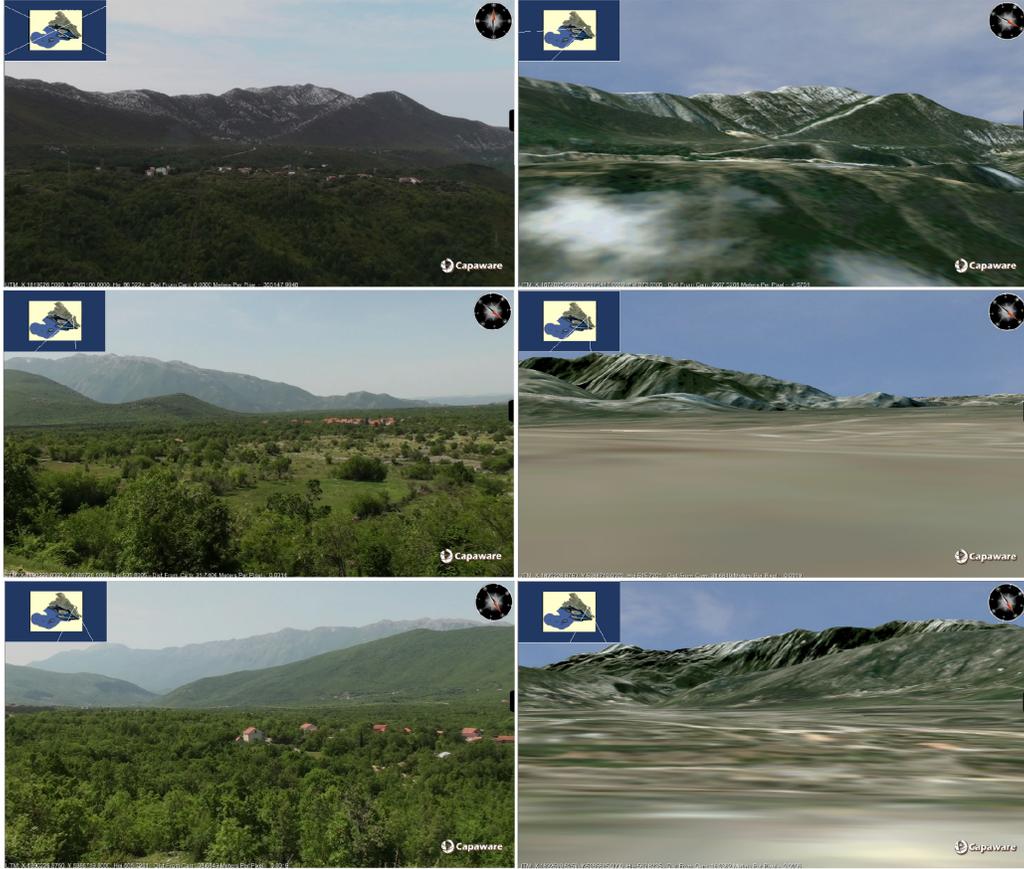


Figure 3: Examples of the scene calibration system that shows the similarities and differences between the real and the virtual terrain. Geographic coordinates are assigned to each pixel on the screen

meteorological and geographical data, we calculate a fire risk index map where fire risk indexes are associated with the geographic coordinates of each point in the image plane.

The pixel coverage and fire risk index maps contain the information about the environment and the current conditions prevailing in the scene. To achieve the desired quality of detection and to ensure that only the latest

and the most detailed information is used, the fire risk data are dynamically calculated at pixel level.

The scene calibration system is connected with a geographic information system that operates within the same geographic region and shares the same map projection and datum. This ensures the compatibility of the data and the interoperability between the two parts of the system. The GIS system automatically computes all the required data before the visual smoke detection algorithm is executed. Recall that the required data are calculated only for the predefined preset positions, which allows the system to operate in real-time.

A more detailed description of the pixel coverage and fire risk index maps is presented in the following subsections.

4.1. Pixel Coverage Map

The geographic coordinates and distances from the camera of each pixel in the image plane could be directly used to produce the depth map. However, the relative distances from the camera stored in the depth map are not always useful, especially if the camera is recording while zoomed in. It is more appropriate to calculate the area of the real world covered by each pixel in the camera image plane.

The actual dimensions of objects in the real world that are visible in the input image are difficult to estimate. In most cases, the information about the third dimension is irretrievably lost. We propose that the size of the object be estimated from its projection on to the plane. This plane has following characteristics: it and the observed object are equal distance from the camera and it is parallel to the camera's image plane.

It can also be concluded that each pixel in the image also represents a space in the real world. This space can be projected on to the described plane. In this way, we can estimate the size of the space visible in the pixel. A map created in this manner, where each pixel represents the size of the space visible in one pixel, is named a pixel coverage map.

Fig. 4 demonstrates the method used to estimate the size of the 3D space projected onto the plane that is parallel to the camera image plane inside one pixel, used for the calculation of the pixel coverage map. The pixel $T_p(x, y)$ shown at coordinates (x, y) displays the three-dimensional space in the vicinity of the point $T_{UTM}(X, Y, Z)$ in the real world. Because elevation data are known for every point of the terrain, the objective is to estimate the dimensions of the space around point T_{UTM} , or in other words the size of the projection of that space onto the plane R containing the point T_{UTM} and parallel to the camera image plane.

Two points are defined, T_{UTM_1} and T_{UTM_2} , that are equally distant from the point T_{UTM} and positioned at the same height $Z = h$ as the point T_{UTM} . Additionally, the points T_{UTM_1} , T_{UTM} and T_{UTM_2} form a straight line segment defined by the distance $2d$. The point T_{UTM_1} is in the image plane shown inside the pixel T_{p1} , and the point T_{UTM_2} is inside the pixel T_{p2} . Therefore, in the image plane a line segment $\overline{T_{p1}T_{p2}}$ is formed, with a defined length $2d_p$.

The lengths d and h are expressed in meters, while length d_p is expressed in the number of pixels. Because d and d_p represent the same length, it is possible to calculate the relationship between those units.

If we make the assumption that the pixels are square, d^2 represents the area of the projected space onto the plane containing point T_{UTM} and parallel

to the camera image plane. Therefore, d^2 represents one value in the proposed pixel coverage map for the observed pixel in the camera image plane.

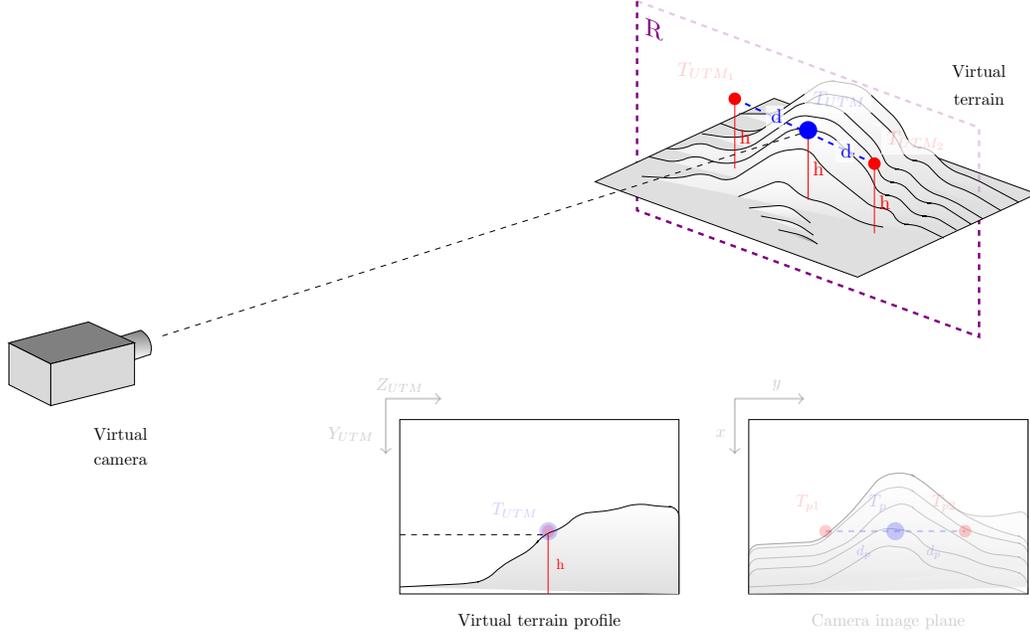


Figure 4: Calculation of pixel coverage values

The pixel coverage map contains information about the real-world coverage area for each pixel in the image expressed in square meters, and should be calculated only once for a single preset position of the camera. Fig. 5 shows a visual representation of the pixel coverage map for the image taken by the real camera shown in Fig. 6(a). The intensity of each pixel inside this visual representation of the pixel coverage map corresponds to the 3D space area that the pixel covers. Note that Fig. 5 is a grayscale image where the intensity of each pixel is limited to 255 values.

In conclusion, terrain that is closer to the camera has a smaller pixel coverage area; the corresponding pixels in Fig. 5 are darker. The pixel

primary sub-indexes representing the fuel moisture, two intermediate sub-indexes representing the rate of spread and fuel consumption and a final index representing fire intensity as an energy output rate per unit length of fire front. The Canadian fire weather index shows the daily variation in temperature, relative humidity, wind speed, and rain. This index is also part of the meteorological fire risk of the European Forest Fires Risk Forecasting System.

Several forest fire indexes are based on estimates of the atmospheric conditions in the vicinity of combustible vegetation or the probability of a fire igniting based on the proximity of dead vegetation and fine fuel moisture. One example of a forest fire index, proposed in [44], takes into account several factors that affect the fire hazards, such as the slope of the terrain or meteorological conditions, emergency response factors, such as watch-tower proximity, and vulnerability factors, such as the population density and the value of forest resources.

In [45, 46] another forest fire index is proposed that is computed from topography, distance from roads and distance from settlements using a combination of remote sensing and GIS data. In [47], a forest fire index is computed as the combination of hazard factors, such as the population density or meteorological conditions, vulnerability factors, such as life and economy vulnerabilities, emergency response and recovery capability factors and exposure factors, such as the residential population or the number of livestock. In [48], a probability-based model is used to estimate the wildfire risk.

Any of the developed systems can be used as the input for the visual smoke detection algorithm. Additionally, we have developed a micro-location fire

risk index [49, 50] that uses several GIS layers to calculate the final fire risk index: climatological and meteorological layers, a vegetation layer, terrain configuration, the history of forest fires, and sociological layers, such as layers based on the human infrastructure and layers based on human activities. Most of the layers used for the micro-location fire risk index are static and only need to be calculated once. On the other hand, meteorological data are collected twice a day, along with a forecast for the next 12 hours, so the corresponding layers should be updated accordingly.

GIS is used to retrieve the fire risk index at specific geographic coordinates representing the point visible at specific pixel coordinates in the image plane. After iterating through all the pixels, a fire risk index map is generated and used as an input for the detection system. The fire risk index depends on dynamic data, meaning that the proposed fire risk index maps are prone to changes and should be calculated several times a day for all preset positions. Fig. 6(b) shows an example of a fire risk index map for the image shown in Fig. 6(a) during a period with a relatively high risk of wildfires.

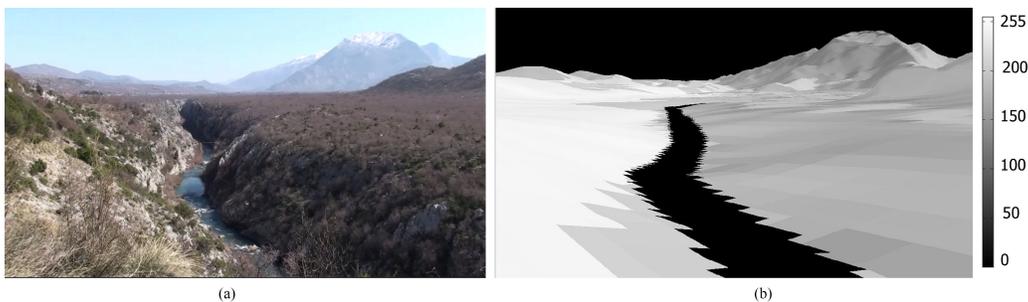


Figure 6: (a) Original image, (b) micro-location fire risk index map for the visible region (darker areas indicate lower fire risk)

5. Detection Parameters Adjustment

In Sect. 2, we have described several phases common to smoke detection processes. Using the spatial and fire risk data given in Sect. 4, we can adjust several standard detection parameters, as well as the stream of input images. These adjustments can be applied to the majority of smoke detection methods because they have similar detection parameters. This can be accomplished with minimal or no modifications to the source code of the algorithms. There are five main improvements based on pixel coverage and fire risk index maps:

- the adjustment of the motion detection sensitivity
- selective blurring of input images based on point distances from the camera
- definition of the minimal candidate region size
- dynamics analysis based on the detected region growth rate
- the final verification using the fire risk index

First, the motion detection sensitivity threshold is adjusted based on the size of the pixel coverage area. It is possible to calculate a single general motion detection sensitivity parameter for the entire scene, but it is better to do this on a pixel level. This allows a more accurate selection of the regions of interest in the observed scene. As mentioned before, motion detection sensitivity is proportional to the distance of the region of interest from the camera, and therefore inversely proportional to the pixel coverage values.

The movement of the objects located at greater distances from the camera is more difficult to detect, and accordingly, the motion detection should be more sensitive in that case. On the other hand, the movement of the objects closer to the camera is accentuated and in many situations the cause of false alarms. In this case the motion detection sensitivity should be notably decreased. We use the information about the distance of the pixel to increase or decrease the sensitivity for that area by a maximum $\pm 10\%$. Fig. 7 shows the successful detection of distant smoke because of the higher motion detection sensitivity for the observed region.

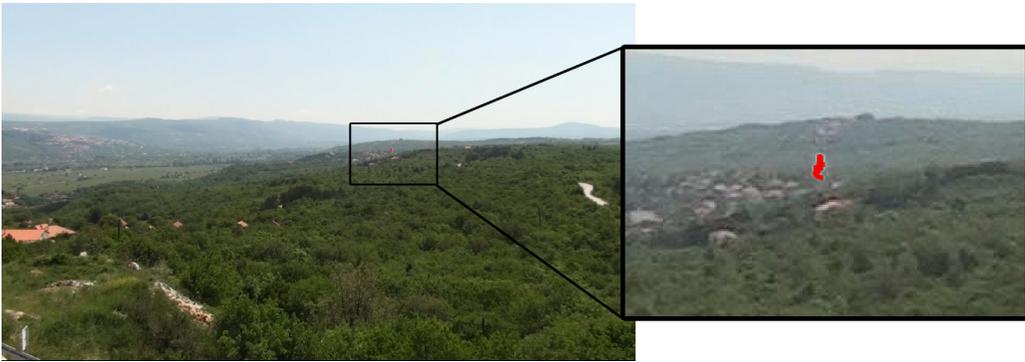


Figure 7: Distant objects and phenomena appear smaller in the image, as their size is inversely proportional to the distance from the camera. Also note that distant movement requires high motion detection sensitivity.

Another important adjustment is modifying the input image stream with selective region blurring. The input is blurred to reduce the unnecessary movement and image noise. The amount of blur is inversely proportional to the distance of the terrain from the camera. We propose motion blurring in the horizontal direction, as it can eliminate most of the unwanted movements that may cause false alarms. As the smoke usually moves upward, it is not

significantly affected by the motion blur in the horizontal direction. An example of image blurring based on pixel coverage map is shown in Fig. 8.



Figure 8: An example of image blurring based on spatial data. (a) Original image, (b) blurred input image. The amount of blur is inversely proportional to the distance from the camera

Another estimated parameter is the threshold value for a minimal region size. Using the pixel coverage map (described in Sect. 4.1), it is possible to estimate the approximate physical size of the smoke region. Although it is not possible to determine the exact size of the smoke region, due to its unpredictable movement and positioning, an approximate estimate is used to improve the candidate region filtering process. Among all the pixels that are detected as smoke, we choose the one with the lowest pixel coverage value. The chosen value, multiplied by the number of connected pixels detected as smoke, represents the real-world size of the detected phenomena. We chose this value to reduce the probability of error because the smoke that rises above the terrain can sometimes be positioned in front of the distant terrain. All detected regions covering less than $30 m^2$ are dismissed as non-smoke regions.

A change in the physical size of the region is an important indicator that allows the dismissal of regions that do not exhibit smoke-like behavior. Smoke detection methods usually define a threshold used to limit the dynamics allowed for the candidate regions. We can estimate the region dynamics threshold by using the pixel coverage map in the same manner as the threshold value for a minimal region size. It is possible to define upper and lower growth rate thresholds taking into account the estimated size of the detected region.

We conducted offline measurements of the rate of spread of smoke in the early wildfire phases. The measurements were conducted on videos recorded with multiple cameras on various locations. Smoke plumes were tracked in terms of size during the incipient phase of wildfire, and we have established that the smoke behaves according to certain rules. The location-scale distribution is the distribution with the best fit on the obtained data that describes the change in the smoke area. We have also observed the standard deviation and the average of the two-dimensional smoke rate of spread in the camera image plane. The offline data show that the average smoke area rate of spread in the image plane is $16.04 \text{ m}^2/\text{s}$ with standard deviation (σ) of $76.33 \text{ m}^2/\text{s}$ in the first 3 minutes after the first occurrence.

During the incipient phase of wildfire, it can be shown that over 99 percent of smoke area change observations fall into the interval defined by $(\mu - 6\sigma, \mu + 6\sigma)$. Because the real smoke region size could be estimated using the proposed pixel coverage map, it is possible to eliminate any phenomena that grows or shrinks faster than the smoke in the incipient phase of wildfire. As an example, it is possible to eliminate significant changes in lighting,

especially intensity changes during sunrise or sunset, shadowing caused by clouds, sunlight artifacts, etc. Details of the experimental analysis of this research are beyond the scope of this paper, but will be published separately in the near future.

Finally, the candidate regions are additionally verified by using the fire risk index map, in this case the micro-location forest fire risk index, to produce a more certain final decision. More information about fire risk index map is given in Sect. 4.2. The fire risk index is taken into account in the last stage of detection, where the final decision about the candidate region is made. The fire risk index for the candidate region is calculated as the mean value of the fire risk indexes of individual pixels contained in the region. After the complete detection and analysis process the final decision about a particular potential alarm has to be made. The decision is made based on different aspects of the analysis of the detected region (depending on the specific detection method) and the information about the fire risk. The fire risk influences the final decision to a certain degree, depending on the preferred tuning of the system and the specific detection method.

6. Evaluation

The proposed improvement of smoke detection systems is evaluated using three smoke detection methods. Every method is tested with and without proposed improvements on a database consisting of 2977 images that are contained in 23 video sequences, of which 16 contain smoke. The images were extracted from the video every 1 second, meaning that the total time span of the footage is approximately 50 minutes. The smoke is visible in its various

forms in 1839 images (a total of approximately 30 minutes), while the rest of the images (1138) are necessary for a quality evaluation and include phenomena that could produce false alarms. Video sequences include phenomena such as clouds and shadowing by clouds, vegetation movement caused by wind, changes in lightning conditions during sunset, and the movements of rivers and sea. Most of the videos from this database are publicly available (http://wildfire.fesb.hr/index.php?option=com_content&view=article&id=65).

Videos were recorded using a “*Sony HDR-CX105E*” video camera with the following values for the intrinsic parameters: $f_x = 1423$, $f_y = 1423$, $u_0 = 600$, and $v_0 = 338$. The size of the input stream image was defined by $width = 1200$, $height = 676$. Considering the quality of the camera, the skew factor and radial lens distortion could be ignored.

The first method (*Method 1*) [2] is a video-based smoke detection method where the smoke detection process is divided into several detection phases. First, the motion detection phase is executed and the moving pixels are extracted based on a background estimate. The background image is iteratively estimated for each frame using the input image and the previous background frame. The next phase is a texture analysis of the detected regions based on local wavelet energy values. The method relies on the premise that the appearance of smoke in the image region smoothes the edges and textures and therefore reduces the amount of high frequencies. The region should also exhibit a decrease in chrominance values, due to the influence of the grayish smoke colors. Another indicator used to determine smoke presence is the flickering appearance around the smoke’s edges. This behavior is evident

from a relatively close distance, but is rather difficult to detect from greater distances. Finally, the shape of the detected region is taken into account, as regions of smoke should have a convex contour. This method is designed primarily for close range detection ($< 100m$), but it can also be used at greater distances ($< 2000m$). The authors have also designed new methods intended for long range smoke detections, such as [51, 52].

The second method (*Method 2*) [3] divides the input image into smaller regions or bins that are represented by the mean value of their constituent pixels. The method takes into account the blue channel of each bin and compares it to the reference image. A substantial increase in the blue channel indicates the possible presence of smoke, and the candidate regions are further analyzed based on smoke dynamics characteristics. For an alarm to be raised, a specified minimum number of bins should be persistently detected over a specified time period. Additionally, a region growth constraint is imposed, permitting a maximum increase in the number of detected bins between consecutive frames. To reduce false alarms, the number of connected components in a single frame is also limited. Finally, an alarm is raised if the detected region satisfies the given constraints during a predefined time period.

The third method (*Method 3*) [53] also consists of several different phases of smoke detection. First, a segmentation and classification phase isolates different image classes, such as sky and water regions, that can be used for the dismissal of some categories of false alarms. Further, a motion detection phase determines the motion regions from the input stream that are analyzed in the following phases. The next step is the chromatic analysis of each

detected region based on color information, where the obtained values are compared to the reference smoke color values. The following phase is the texture analysis of the candidate region based on wavelet information. The change in texture and the loss in the high frequency range is used as an indicator of smoke presence in the region. Further, dynamic characteristics of the candidate regions are examined for smoke-like behavior over the specified number of frames. Finally, an alarm is raised when the detected regions are confirmed in each phase of detection and exhibit static and dynamic smoke characteristics.

For each method, a revised version is designed based on parameters and selective blurring of input stream images at pixel level that is proposed in the previous section. However, selective blurring is not implemented for *Method 2* because the method performs binning of input images; therefore, additional noise reduction is not necessary.

The evaluation of all methods is performed using the evaluation measures for smoke detection algorithms described in [54]. Examples of detection images for all improved methods are shown in Fig. 9.

The evaluation measures are divided into two categories: global and local measures. Global measures evaluate algorithm performance based on results where the smallest detection units are images, and evaluation is treated as a binary classification problem. There are four main measures describing different aspects of detection quality: measures for correct detections (cd), correct rejections (cr), false alarms (fa) and missed detections (md). They are defined as;

$$cd = \frac{TP}{TP + FN} \quad (13)$$

$$cr = \frac{TN}{TN + FP} \quad (14)$$

$$fa = \frac{FP}{TN + FP} \quad (15)$$

$$md = \frac{FN}{TP + FN} \quad (16)$$

where TP represents the number of true positive detections, TN represents the number of true negative detections, FP represents the number of false positive detections, and FN represents the number of false negative detections. In the case of global measures, the values TP , TN , FP and FN represent true or false detections or rejections, where the detection units are images.

On the other hand, local measures regard individual pixels as smallest detection units and address the quality of detection on this level. When comparing several detection methods, the overall quality is best described using the global measures because the system alarm is raised on the global scale. The exact location of the smoke region in the image, evaluated by the local measures, is used as a secondary indicator in the event the global values are similar.

First, general results based on all sequences are presented for each method in Table 1, including the information about the farthest distance where smoke was detected.

An improvement in the reliability of detection is evident for all the methods. Another important aspect is the noticeable improvement in detection range for all the methods. The detection range is the maximum distance at

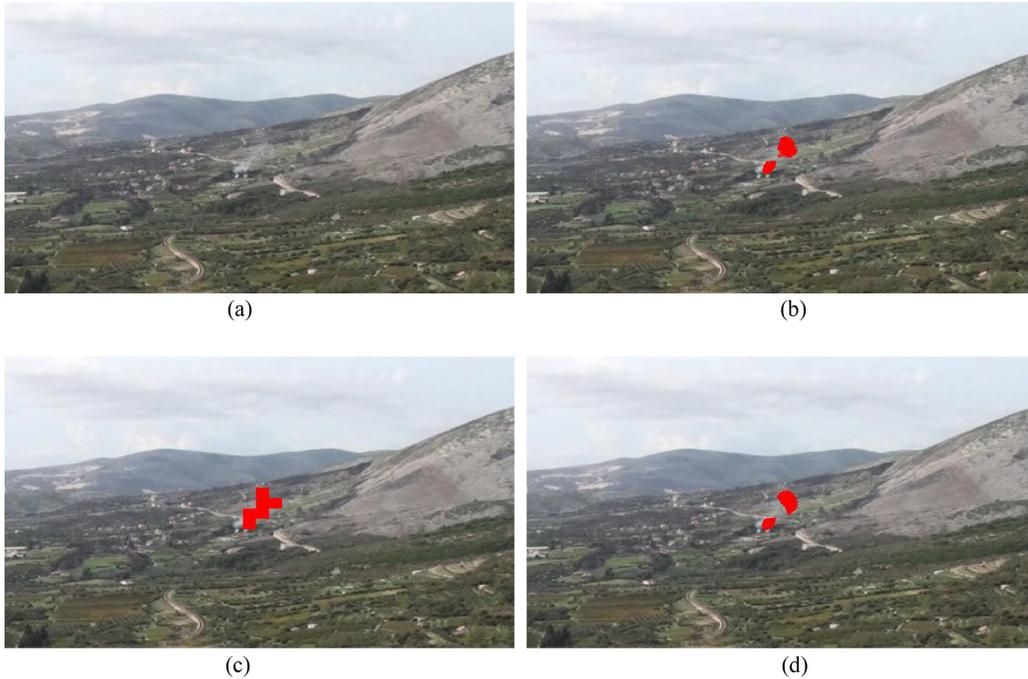


Figure 9: (a) Input image with visible smoke, (b) image with smoke detected by *Method 1* (improved), (c) image with smoke detected by *Method 2* (improved), (d) image with smoke detected by *Method 3* (improved))

which smoke is detected in a given set of sequences. The improvement in the detection range is the result of detection sensitivity adjustment based on the spatial data and the pixel coverage map. This allows for increased detection sensitivity for areas that are located at greater distances from the camera.

It is important to emphasize that only sequences captured with the default zoom level were taken into consideration for these experiments. Additional improvement in detection range might be achieved by introducing a zoom factor into the input stream, but it was not the topic of this research.

The global measures for all methods are presented in Table 2. The results

Table 1: Detection results for all sequences

	Smoke sequences detected	Missed smoke sequences	Farthest detected smoke distance (m)
<i>Method 1</i> - standard	13	3	2765
<i>Method 1</i> - improved	15	1	4255
<i>Method 2</i> - standard	13	3	2765
<i>Method 2</i> - improved	15	1	4255
<i>Method 3</i> - standard	14	2	2765
<i>Method 3</i> - improved	16	0	9210

show an increase in correct detections for all methods, while maintaining or reducing the number of false alarms. The results for correct rejections and missed detections indicate the same trend because they are complementary to false alarms and correct detections, respectively.

The authors in [54] propose the usage of observer quality graphs that show the values of the specific measure for all the images in the collection, sorted according to increasing measure values. In the graphs, the y axis represents the value of the specific measure, and the x axis represents the image rank, or the ordinal number of the image in the sorted sequence. Figs. 10, 11 and 12 show the observer quality graphs for the local measures: correct detections (cd), false alarms (fa) and the Matthews correlation coefficient (mcc), respectively. The Matthews correlation coefficient [55] is defined by:

Table 2: Global measures

	Correct detections	Correct rejections	False alarms	Missed detections
<i>Method 1</i> - standard	0.444	0.968	0.032	0.556
<i>Method 1</i> - improved	0.745	0.999	0.001	0.255
<i>Method 2</i> - standard	0.375	1.000	0.000	0.625
<i>Method 2</i> - improved	0.591	0.999	0.001	0.409
<i>Method 3</i> - standard	0.701	0.939	0.061	0.299
<i>Method 3</i> - improved	0.776	0.984	0.016	0.224

$$mcc = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \quad (17)$$

and is frequently used as a measure of quality of binary classifications. The results for the local Matthews correlation measure (mcc) are shown in Fig. 12. This coefficient takes into account true, false positive and negative detections in a balanced manner, so the measure can be used even if the classes are of very different sizes.

The results for the local cd measures show an increase in correct detections for all improved methods in Fig. 10. However, there is also an increase in false alarms on the local levels as shown in Fig. 11. This effect is the result of input image blurring, which decreases the false alarm rate (the value of the fa measure) on the global level and increases the false alarm rate on pixel level. The blurring of the input image makes the method less sensitive to the noise in the environment but also makes it more difficult to extract

the correct smoke region boundaries. Because of this effect, the regions are segmented beyond the real boundaries of the moving object, resulting in an increase in false positive pixels. The reliability of the overall detection is not, however, affected by this effect.

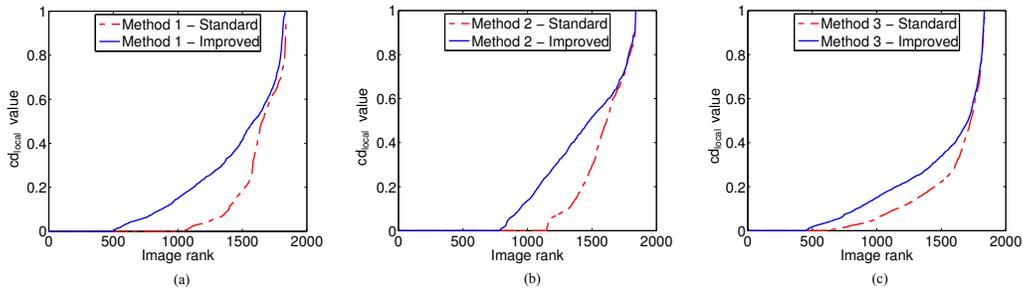


Figure 10: Correct detections - (a) *Method 1*, (b) *Method 2* and (c) *Method 3*

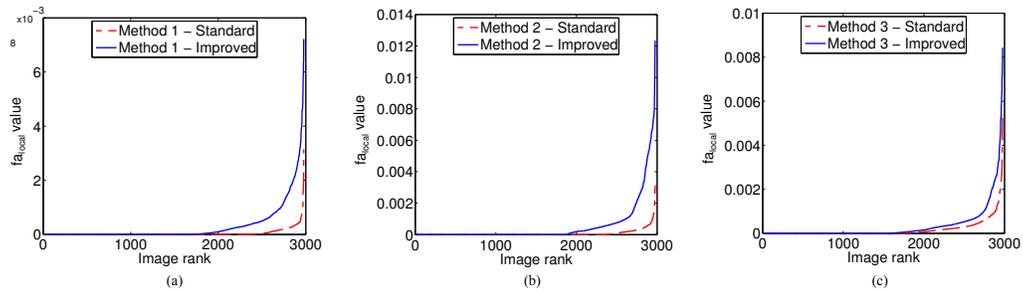


Figure 11: False alarms - (a) *Method 1*, (b) *Method 2* and (c) *Method 3*

Mcc returns a value in the interval $[-1, 1]$, where 1 represents a perfect prediction. As seen in Fig. 12, the evaluation results for a local mcc measure show a considerable increase in the Matthews correlation coefficient values for all improved methods.

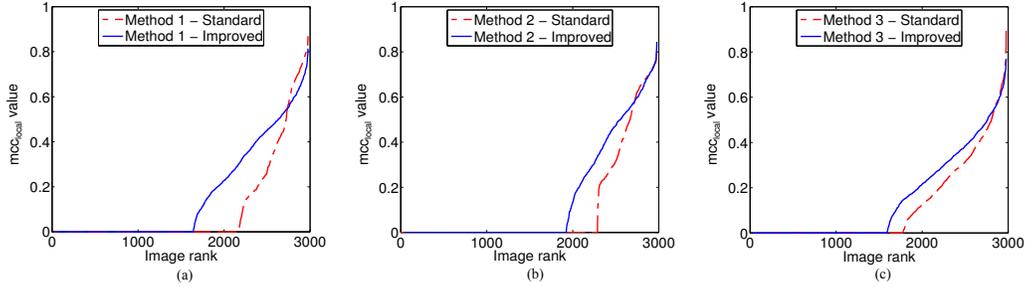


Figure 12: Matthews correlation coefficients - (a) *Method 1*, (b) *Method 2* and (c) *Method 3*

7. Conclusions

In this paper, we presented an improvement in the field of visual smoke detection based on the adaptive adjustment of smoke detection parameters and the modification of the images in the input stream. This improvement was achieved using GIS and GIS-based augmented reality to provide a more robust and reliable smoke detection. Most of the existing smoke detection algorithms are designed for general use in which the user manually adjusts the detection sensitivity parameters for the entire scene. This approach directly affects the rate of both false alarms and correct detections. We have shown that the existing methods could be improved by introducing scene-specific parameter adjustments based on spatial and fire risk data for each pixel of the input image. This improvement was achieved by integrating computer vision and augmented reality techniques.

Five different aspects of the improvement to visual smoke detection are proposed in this article: the adjustment of motion detection sensitivity, selective blurring of input images based on point distance from the camera,

definition of the minimal candidate region size, dynamics analysis based on detected region growth rate and fire risk assessment using the fire risk index.

The evaluation conducted has shown improvement in the quality of proposed smoke detection methods for different detection aspects. The false alarms for standard methods were primarily caused by such phenomena as vegetation movement caused by wind, clouds and cloud shadows, the movements of rivers and sea or changes in lightning conditions during sunset, all of which were included in the video database used for evaluation. As reflected in the results of the evaluation, the revised methods were less sensitive to these phenomena and have shown improvement in false alarm rate, correct detection rate and detection range.

The most evident improvement on global level was noted for *Method 1*, where the *cd* measure increased by more than 0.3, while the number of false alarms was reduced to the minimum level. The difference in quality between the standard and improved methods was least noticeable for *Method 3*. However, note that the standard *Method 3* already provides good results (approximately 0.7 for *cd* measure). At local level, the improvements are less obvious. However, based on the Matthews correlation coefficient, an improvement is evident for all three methods.

Smoke detection system must be executed in real time. Our system performs in real-time if predefined preset positions are used, and all required data are calculated prior to detection. One of the primary drawbacks of our system is the inability to operate in real time when the preset positions are not predefined. The time required to calculate the pixel coverage and fire risk index maps is several minutes, depending on the image resolution.

Additional efforts are necessary in order to implement the proposed improvements into manual operation of smoke detection systems where the camera movement is unpredictable.

In conclusion, we demonstrated that the overall number of sequences where smoke was detected was increased by the proposed improvements without compromising the robustness and reliability of the system. However, the operation of the improved versions is still not ideal, and there is still room for further work and improvement in all aspects of smoke detection.

8. References

- [1] D. Stipaničev, M. Štula, D. Krstinić, L. Šerić, T. Jakovčević, M. Bugarić, Advanced automatic wildfire surveillance and monitoring network, in: Proceedings of the VI Int. Conf. on Forest Fire Research, 2010, pp. 1–15.
- [2] B. Toreyin, Y. Dedeoglu, A. Cetin, Wavelet based real-time smoke detection in video, in: Proceedings of the 13th European Signal Processing Conference EUSIPCO, Vol. 20, 2005, pp. 255–26.
- [3] J. Fernández-Berni, R. Carmona-Galán, L. Carranza-González, A vision-based monitoring system for very early automatic detection of forest fires, in: Proceedings of the First International Conference on Modelling, Monitoring and Management of Forest Fires (FIVA 2008), 2008, pp. 161–170.
- [4] D. Stipaničev, L. Šerić, M. Braović, D. Krstinić, M. Štula, M. Bugarić, Vision Based Wildfire and Natural Risk Observers, in: Proceedings of the 3rd International Conference on Image Processing Theory, Tools

- and Applications, OS1: Special session on Image Processing for Natural Risks (IPNR), 2012.
- [5] R. D. Labati, A. Genovese, V. Piuri, F. Scotti, Wildfire Smoke Detection Using Computational Intelligence Techniques Enhanced With Synthetic Smoke Plume Generation, *IEEE Trans. Syst. Man. Cybern.: Syst.* 43 (4) (2013) 1003–1012.
 - [6] M. Štula, D. Krstinić, L. Šerić, Intelligent forest fire monitoring system, *Inf. Syst. Front.* 14 (3) (2012) 725–739.
 - [7] A. Benazza-Benyahia, N. Hamouda, F. Tlili, S. Ouerghi, Early smoke detection in forest areas from DCT based compressed video, in: *Proceedings of the 20th European Signal Processing Conference (EUSIPCO)*, 2012, pp. 2752–2756.
 - [8] Y. Zhao, W. Lu, Y. Zheng, J. Wang, An early smoke detection system based on increment of optical flow residual, in: *Proceedings of International Conference on Machine Learning and Cybernetics*, Vol. 4, 2012, pp. 1474–1479.
 - [9] E. Zervas, A. Mpimpoudis, C. Anagnostopoulos, O. Sekkas, S. Hadjiefthymiades, Multisensor data fusion for fire detection, *Inf. Fusion* 12 (3) (2011) 150–159.
 - [10] Z. Xu, J. Xu, Automatic Fire Smoke Detection Based on Image Visual Features, in: *Proceedings of the International Conference on Computational Intelligence and Security Workshops, CISW*, 2007, pp. 316–319.

- [11] B. U. Toreyin, Y. Dedeoglu, A. E. Cetin, Contour based smoke detection in video using wavelets, in: Proceedings of the 14th European Signal Processing Conference EUSIPCO, 2006, pp. 1–5.
- [12] P. Piccinini, S. Calderara, R. Cucchiara, Reliable smoke detection in the domains of image energy and color, in: Proceedings of the 15th IEEE International Conference on Image Processing, 2008, pp. 1376–1379.
- [13] A. Genovese, R. D. Labati, V. Piuri, F. Scotti, Wildfire smoke detection using computational intelligence techniques, in: Proceedings of the IEEE International Conference on Computational Intelligence for Measurement Systems and Applications (CIMSA), 2011, pp. 1–6.
- [14] D. Kim, Y.-F. Wang, Smoke Detection in Video, in: Proceedings of WRI World Congress on Computer Science and Information Engineering, 2009, pp. 759–763.
- [15] A. Ochoa-Brito, L. Millan-Garcia, G. Sanchez-Perez, K. Toscano-Medina, M. Nakano-Miyatake, Improvement of a Video Smoke Detection Based on Accumulative Motion Orientation Model, in: Proceedings of the Electronics, Robotics and Automotive Mechanics Conference (CERMA) IEEE, 2011, pp. 126–130.
- [16] T. X. Truong, J.-M. Kim, An early smoke detection system based on motion estimation, in: Proceedings of the International Forum on Strategic Technology (IFOST), 2010, pp. 437–440.
- [17] C. Ho, T. Kuo, Real-time video-based fire smoke detection system, in:

Proceedings of the International Conference on Advanced Intelligent Mechatronics, 2009, pp. 1845–1850.

- [18] L. Ma, K. Wu, L. Zhu, Fire Smoke Detection in Video Images Using Kalman Filter and Gaussian Mixture Color Model, in: Proceedings of the International Conference on Artificial Intelligence and Computational Intelligence, Vol. 1, 2010, pp. 484–487.
- [19] T.-H. C.-H. Chen, Y.-H. Yin, S.-F. Huang, Y.-T. Ye, The Smoke Detection for Early Fire-Alarming System Base on Video Processing, in: Proceedings of the IHH-MSP '06. International Conference on Intelligent Information Hiding and Multimedia Signal Processing, Vol. 0, 2006, pp. 427–430.
- [20] D. Krstinić, D. Stipaničev, T. Jakovčević, Histogram-based smoke segmentation in forest fire detection system, *Inf. Technol. Control* 38 (3) (2009) 237–244.
- [21] Y. Chunyu, F. Jun, W. Jinjun, Z. Yongming, Video Fire Smoke Detection Using Motion and Color Features, *Fire Technol.* 46 (3) (2009) 651–663.
- [22] Y. Cui, H. Dong, An Early Fire Detection Method Based on Smoke Texture Analysis and Discrimination, in: Proceedings of . the Congress on Image and Signal Processing (CISP '08), 2008, pp. 95–99.
- [23] Hidenori Maruta, Yasuharu Kato, Akihiro Nakamura, Fujio Kurokawa, Smoke detection in open areas using its texture features and time series

- properties, in: Proceedings of the IEEE International Symposium on Industrial Electronics (ISIE), 2009, pp. 1904–1908.
- [24] Y. Chunyu, Z. Yongming, F. Jun, W. Jinjun, Texture Analysis of Smoke for Real-Time Fire Detection, in: Proceedings of WCSE '09. Second International Workshop on Computer Science and Engineering, 2009, pp. 511–515.
- [25] R. M. Haralick, K. Shanmugam, I. Dinstein, Textural Features for Image Classification, *IEEE T. Syst. Man. Cyb.* 3 (6) (1973) 610–621.
- [26] A. Buch, W. W. Boles, Texture classification using multiple wavelet analysis, in: Proceedings of the Sixth Digital Image Computing: Techniques and Applications Conference, 2002, pp. 341–345.
- [27] B. Kara, N. Watsuji, Using wavelets for texture classification, in: Proceedings of the International Conference on Signal Processing, 2003, pp. 920–924.
- [28] W. Meng-Yu, H. Ning, L. Qin-Juan, A Smoke Detection Algorithm Based on Discrete Wavelet Transform and Correlation Analysis, in: Proceedings of the Fourth International Conference on Multimedia Information Networking and Security (MINES), 2012, pp. 281–284.
- [29] A. Rafiee, R. Dianat, M. Jamshidi, R. Tavakoli, S. Abbaspour, Fire and smoke detection using wavelet analysis and disorder characteristics, in: Proceedings of the 3rd International Conference on Computer Research and Development (ICCRD), 2011, pp. 262–265.

- [30] D. Wu, N. Wang, H. Yan, Smoke detection based on multi-feature fusion, in: Proceedings of the 5th International Congress on Image and Signal Processing (CISP), 2012, pp. 220–223.
- [31] J. Y. Kwak, B. C. Ko, J.-Y. Nam, Forest Smoke Detection Using CCD Camera and Spatial-temporal Variation of Smoke Visual Patterns, in: Proceedings of the Eighth International Conference on Computer Graphics, Imaging and Visualization (CGIV), 2011, pp. 141–144.
- [32] B. Ko, J.-Y. Kwak, J.-Y. Nam, Wildfire smoke detection using temporospatial features and random forest classifiers, *Opt. Eng.* 51 (1) (2012) 017208–1–017208–10.
- [33] J. Park, B. Ko, J.-Y. Nam, S. Kwak, Wildfire smoke detection using spatiotemporal bag-of-features of smoke, in: Proceedings of the IEEE Workshop on Applications of Computer Vision (WACV), 2013, pp. 200–205.
- [34] J. Yang, F. Chen, W. Zhang, Visual-Based Smoke Detection Using Support Vector Machine, in: Proceedings of the Fourth International Conference on Natural Computation, 2008. ICNC '08., 2008, pp. 301–305.
- [35] S. Brooks, J. L. Whalley, Multilayer hybrid visualizations to support 3D GIS, *Comput. Environ. Urban.* 32 (4) (2008) 278–292.
- [36] K. Appleton, A. Lovett, G. Sünnerberg, T. Dockerty, Rural landscape visualisation from GIS databases: a comparison of approaches, options and problems, *Comput. Environ. Urban.* 26 (23) (2002) 141–162.

- [37] J. Döllner, K. Hinrichs, An object-oriented approach for integrating 3D visualization systems and GIS, *Comput. Geosci.* 26 (1) (2000) 67–76.
- [38] M. Castrillón, P. A. Jorge, I. J. López, A. Macías, D. Martín, R. J. Nebot, I. Sabbagh, J. Sánchez, A. J. Sánchez, J. P. Suárez, Others, Wildfire Prevention and Management in a 3D Virtual Environment, in: *Proceedings of STATGIS'09: Geoinformatics for Environmental Surveillance*, 2009, pp. 1–6.
- [39] J. Heikkila, Geometric camera calibration using circular control points, *IEEE T. Pattern Anal.* 22 (10) (2000) 1066–1077.
- [40] M. Li, Correspondence analysis between the image formation pipelines of graphics and vision, in: *Proceedings of the IX Spanish Symposium on Pattern Recognition and Image Analysis*, 2001, pp. 187–192.
- [41] H. Ning, Y. Guang-qun, W. Yuan-yuan, Design of automatic forest fire positioning system based on video monitoring system, in: *Proceedings of the Second IITA International Conference on Geoscience and Remote Sensing (IITA-GRS)*, Vol. 2, 2010, pp. 532–535.
- [42] J. San-Miguel-Ayanz, P. Barbosa, G. Schmuck, G. Liberta, E. Schulte, Towards a coherent forest fire information system in Europe: the European Forest Fire Information System (EFFIS), in: *Forest fire research and wildland fire safety: Proceedings of IV International Conference on Forest Fire Research 2002 Wildland Fire Safety Summit*, 2002, pp. 5–16.
- [43] C. E. Van Wagner, S. canadien des Forêts, Structure of the Canadian forest fire weather index, Environment Canada, Forestry Service, 1974.

- [44] C. Gai, W. Weng, H. Yuan, GIS-Based Forest Fire Risk Assessment and Mapping, in: Proceedings of the Fourth International Joint Conference on Computational Sciences and Optimization (CSO), 2011, pp. 1240–1244.
- [45] X. Dong, D. Li-min, S. Guo-fan, T. Lei, W. Hui, Forest fire risk zone mapping from satellite images and GIS for Baihe Forestry Bureau, Jilin, China, *J. Forest. Res.* 16 (3) (2005) 169–174.
- [46] E. Erten, V. Kurgun, N. Musaoglu, Forest Fire Risk Zone Mapping From Satellite Imagery And GIS: A Case Study, *Int. J. Appl. Earth Obs.* 4 (2002) 1–10.
- [47] T. Zhijun, Z. Jiquan, L. Xingpeng, GIS-based risk assessment of grassland fire disaster in western Jilin province, China, *Stoch. Env. Res. Risk A.* 23 (4) (2009) 463–471.
- [48] H. K. Preisler, D. R. Brillinger, R. E. Burgan, J. W. Benoit, Probability based models for estimation of wildfire risk, *Int. J. Wildland Fire* 13 (2) (2004) 133–142.
- [49] M. Bugarić, T. Jakovčević, D. Stipaničev, Automatic adjustment of detection parameters in forest fire video monitoring system, in: Proc. of 32. Int. Conference MIPRO 2009, 2009, pp. 270–275.
- [50] D. Stipaničev, B. Marin, M. Štula, L. Šerić, Holistic Forest Fire Prevention Model of Split and Dalmatia County, Part III Microlocation Fire Risk Index Calculation (in Croatian), Tech. rep., FESB, Split (2008).

- [51] O. Günay, B. U. Töreyn, K. Köse, A. E. Çetin, Entropy functional based online adaptive decision fusion framework with application to wildfire detection in video, *IEEE T. Image Process.* 21 (5) (2012) 2853–2865.
- [52] O. Günay, B. U. Töreyn, A. E. Çetin, Online Adaptive Decision Fusion Framework Based on Projections onto Convex Sets with Application to Wildfire Detection in Video, *Opt. Eng.* 50 (7) (2011) 077202–077202–12.
- [53] T. Jakovčević, D. Stipaničev, D. Krstinić, Visual spatial-context based wildfire smoke sensor, *Mach. Vision. Appl.* 24 (4) (2012) 707–719.
- [54] T. Jakovčević, L. Bodrožić, D. Stipaničev, D. Krstinić, Wildfire smoke-detection algorithms evaluation, in: *Proceedings of the VI International Conference on Forest Fire Research*, 2010, pp. 1–12.
- [55] B. W. Matthews, Comparison of the predicted and observed secondary structure of T4 phage lysozyme, *Biochim. Biophys. Acta* 405 (2) (1975) 442–451.