

Intelligent forest fire monitoring system

Maja Stula · Damir Krstinic · Ljiljana Seric

Published online: 30 March 2011
© Springer Science+Business Media, LLC 2011

Abstract This paper presents iForestFire, an Environmental Monitoring Information System for forest fire protection. The system is composed of several components, each having a particular function. Automatic fire detection is a crucial component of the system. It is based on various complex image processing algorithms. Complexity of the system also emerges from integration, based on multi agent technology, of different environment information. The presented system contributes to the environment protection and is in use in Croatia for several years.

Keywords Environmental Monitoring Information Systems · Software architecture · Image processing · Data organization · Fire protection

1 Introduction

Environment protection is one of the most important tasks for the human kind. Huge efforts in all areas of a society are directed to the preservation of natural heritage as much as possible. Information and communication technologies (ICT) take up environment protection to a higher level. The terms Environmental Information Systems (EIS) and Environmental Monitoring Information Systems (EMIS)

have arisen (Nesis project report 2010). Here, we use the term Environmental Monitoring Information System for an information system integrated into the environment for the purpose of gathering, processing and storing environmental information. Intelligent Forest Fire Monitoring System, implemented under the name iForestFire®, is a typical EMIS system. The term intelligent in the system title comes from integrating intelligent methods and technologies into this information system. For example, we have defined rule-based system for environment self-protection ability introduced by an advanced sensor network implemented with a multiagent system integrated into the environment (Seric 2010).

This paper presents iForestFire system development and architecture. The system is described from the functional point of view, identifying parts of the system with the particular function described in detail. iForestFire is designed and implemented primarily to help protection of a forest environment from a fire. Forest fires represent a constant threat to ecological systems, infrastructure, and human lives. Croatia belongs to countries with the enhanced summer forest fire risk particularly in the Dalmatian coast and islands. According to Croatian Ministry of the Interior, from 2000 to 2006, there were 33,234 fires of open space (Ministry of the Interior statistical report 2010). During the same period, burned area was 3,586.34 km². Considering Croatian terrestrial area of 56,542 km² in 6 years, more than 6% of the land was burnt. Major tragedy occurred on August 30 2007 when in just one fire on the island Veliki Kornat, 12 fire fighters were killed.

An effective way to minimize the damage caused by forest fires is early fire detection enabling fast fire fighters' reaction. Human surveillance is a traditional approach not just for forest fire detection but for the detection of different natural phenomena. Forest fire detection in Croatia is legislatively listed and organized by different local and national organizations like Croatian Forests, city councils,

M. Stula (✉) · D. Krstinic · L. Seric
Faculty of Electrical Engineering, Machine Engineering
and Naval Architecture, University of Split,
R. Boskovicova bb,
21000 Split, Croatia
e-mail: maja.stula@fesb.hr

D. Krstinic
e-mail: dkrst@fesb.hr

L. Seric
e-mail: ljiljana@fesb.hr

etc. It is realized through 24 h observation by human observers located on monitoring spots. Human observers are usually equipped only with standard binoculars and communication equipment and their observation area is only the area covered by their sight of view. Installation of remotely controlled video cameras on monitoring spots connected to a monitoring center equipped with adequate video presentation and video storing devices places a human observer into monitoring center. The human observer is capable of monitoring a wider area covered by several video monitoring field units. One human observer can now monitor an area that previously needed several observers, so human power can be transferred to other tasks, like preventive measures. Information and data can be stored for later analysis, which is also quite useful. The next reason for implementing EIS is the possibility of automatization of different tasks. In this paper, the main emphasis is on the automatic fire detection, but many other tasks can be automatized using different technical solutions. Automatic fire detection facilitates monitoring of several cameras with attracting attention of a human observer to possible suspicious situations.

Currently, we are working on one more automatized task on integrating fire simulation with geographical information that would find all cameras on all monitoring spots that can monitor a fire and would automatically direct all those cameras to the fire. Automatization of any task reduces human effort, usually speeds up a process, and generally increases any system's quality. This also applies for EMIS.

2 System development

The development of the iForestFire system began in 2003 with a small team of four people (Prof. Stipanicev, Stula, Krstinic, and Seric). It was partially founded by the Croatian Ministry of Science through technology project No. TP 010023-02 and scientific project No.0230028 and No.023-0232005-2003. The idea was to develop a system with basic functionality of automatic fire detection and to provide it to fire fighters in Split, a town in Dalmatia County, because we established cooperation with them. This cooperation was the result of a productive and intensive communication with Croatia main fire fighter assistant, Mr. Tomislav Vuko (Stipanicev et al. 2006).

To develop the system, we had to address several issues. The main ones are explained here. First, we had to define complete hardware infrastructure. What cameras should we use? What kind of network should we build to communicate the acquired information? Where should we place monitoring locations? How would we process acquired information?

The second issue was that we did not have user requirements. Fire fighters did not know what this new information system should provide to them. We practically

defined system requirements in the initial system version and used feedback from fire fighters to improve the iForestFire and generate new versions. Functional requirements were defined investigating current systems with a similar function like FireWatch (FireWatch 2010) and Forest Fire Finder (Forest Fire Finder 2010). For nonfunctional requirements, we strongly relied on knowledge gained in previous projects. For example, previous projects were the main reason why we decided on web-based information system. Such information systems can easily be provided to the public and connected with public information sources (web portals, social networks, etc.), which is very important for environment protection and environment hazards notification systems (Puras and Iglesias 2009).

In this situation, we could not use heavily formal methods for information system development, like, for example, waterfall development. The best candidate for the system development method was one of the agile development methods. We have chosen extreme programming (XP) (Beck and Andres 2005). Why extreme programming? The first reason was a small and very tightly connected team (and still is, since we continue to improve the system and generate new versions). We are more than coworkers. We are in contact every day. During the project, we would sit down together for hours to solve a problem.

For example, how would a module for image processing, running on a server, provide information to a user interface, running on a client computer, on a possible alarm? Agreed solution was to put the information from image processing module to a database. The part of the user interface, developed as an Ajax component, is querying database every 10 s. When new information on possible alarm appears in the database, the Ajax component pushes alarm information in a pop-up window to a user (Fig. 1).

Second reason was the tight connection with users. After user training, the system was delivered to fire fighters.



Fig. 1 Alarm pop-up window

System acceptance testing was very successful. Users would call us whenever something in the system was missing or was poorly usable according to them. For example, alarms were raised in a pop-up window with image on which alarm was suspected (Fig. 1). Users suggested that alarm information would be more usable if the system highlights just the part of the image that was suspicious. We redesigned the module for automatic detection to produce a new image with highlighted areas in the case of an alarming situation (Fig. 2).

Second, a very important new requirement, that has risen when fire fighters got the system, was the possibility to manually control system cameras and other sensors. We did not include this functionality initially. We were developing the system for the automatic fire detection but when the users experienced the tele-presence with high zooming possibility, they asked for system manual control without automatic fire detection.

To summarize, iForestFire system was developed with a bit modified XP method. We could not always program in pairs, because there was a lot of work and it was a team with only four of us. Deployment was also addressed, although XP does not cover system deployment, because we had to round entire SDLC (System Development Life Cycle) (Fig. 3).

A lot of effort has been made in forest fire prevention and since 2003 we have intensively worked on the development of iForestFire system. At this moment, there are 12 iForestFire systems installed (Stipanicev et al. 2009b). Seven systems are installed in Istra County covering the whole county. Rest of the installed systems are along Adriatic coast in national parks Mljet and Paklenica, and nature parks Biokovo, Telascica, and Vransko jezero. The first system was installed in the

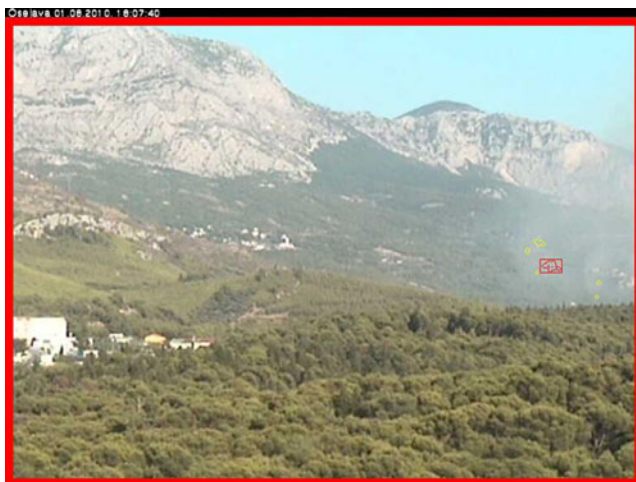


Fig. 2 New alarm pop-up window adjusted according to the user requirement

national park Paklenica in the summer of 2006 and is up and running since then.

3 System architecture

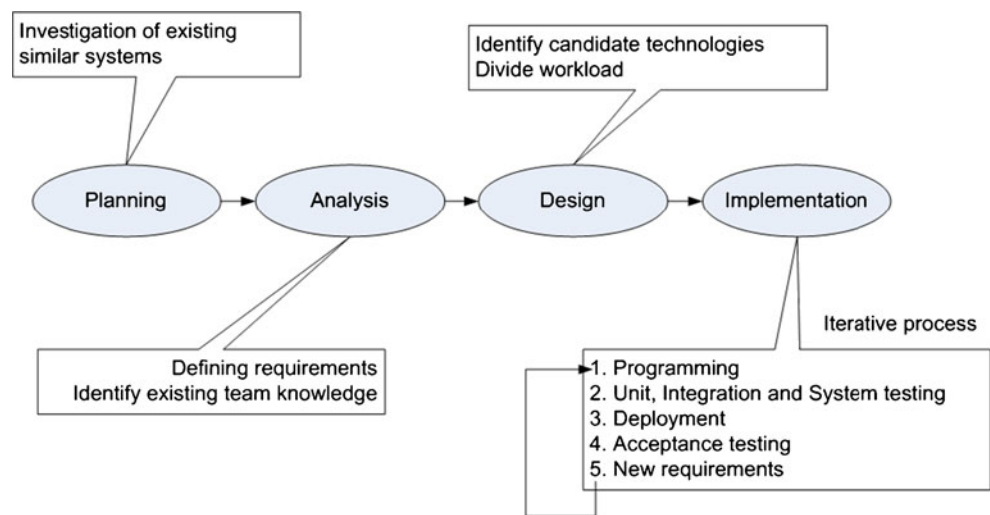
iForestFire is an intelligent and integral video-based monitoring system for the early detection of forest fires.

iForestFire is an intelligent system because in its automatic mode, the forest fire detection is based on various autonomous advanced image processing, image analyses and image understanding algorithms. Algorithms include lot of procedures derived from the fields of Artificial Intelligence and Computational Intelligence. iForestFire software organization is based on agent architecture. Intelligent software agents are responsible for image collecting, image and data storing, sensors integrity testing, image preprocessing, image postprocessing, pre-alarm, and alarm generation. Forest fires are detected in incipient stage using advanced image processing and image analyses methods. Intelligent fire recognition algorithms analyze images automatically, trying to find visual signs of forest fire, particularly forest fire smoke during the day and forest fire flames during the night. If something suspicious is found, pre-alarm is generated and appropriate image parts are visibly marked. The operator inspects suspicious image parts and decides whether it is really the forest fire or not.

iForestFire is an integral system because it consists of independent software components (image processing, meteorological data collection and processing, system georeferencing, fire spreading simulation,...) integrated into one product called iForestFire. Here, the term software component is used very loosely to mean a functionally recognizable piece of software and not in the sense of the more strict definition by Szyperski (Szyperski 1998).

Functionally recognizable piece of software is a piece of software providing a particular function. Components can function independently, providing particular function, but integrated provide more user-friendly, more efficient, and intelligent system. For example, the collected real meteorological data are used as an input for fire spreading simulation in iForestFire instead of artificially generated data, when spreading simulation component works independently.

iForestFire is a Web Information System (WIS) (Isakowitz et al. 1998). All iForestFire components can be reached and administrated through dynamic and interactive web pages. Real-time video and meteorological data are shown on web pages together with GIS data and interface for pan/tilt/zoom camera control when the system is in manual mode. The function of all components is presented as a part of web application to a user, although some components like automatic fire detection are not web-based application.

Fig. 3 iForestfire SDLC

iForestFire UML deployment diagram is shown in Fig. 4.

Deployment diagram is quite complex because the system is complex. It could be more detailed but then it would be even harder to track the system from the diagram. Figure 5 shows the system architecture with block diagram containing main software components on high functional level that depicts iForestFire system and is easier to understand.

The system hardware architecture is based on field units and a central processing unit. The field unit includes the day and night, pan/tilt/zoom-controlled IP-based video camera, and an IP-based mini meteorological station connected by wired or wireless LAN to a central processing unit where all analysis, calculation, presentation, image, and data archiving is done.

iForestFire system architecture from the functional point of view (ANSI/IEEE 1471–2000, Recommended Practice for Architecture Description of Software-Intensive Systems) identifies main system functionalities as follows:

- Data gathering from video and meteo sensors in real time
- Automatic fire detection from acquired video data
- Data archiving for later retrieval
- Geolocation information system for positioning sensors in space
- Fire risk and spreading simulation

Our next sections provide description of the system's functional components. The most important component is automatic fire detection, so this component is explained in detail.

3.1 Data gathering from video and meteo sensors in real time

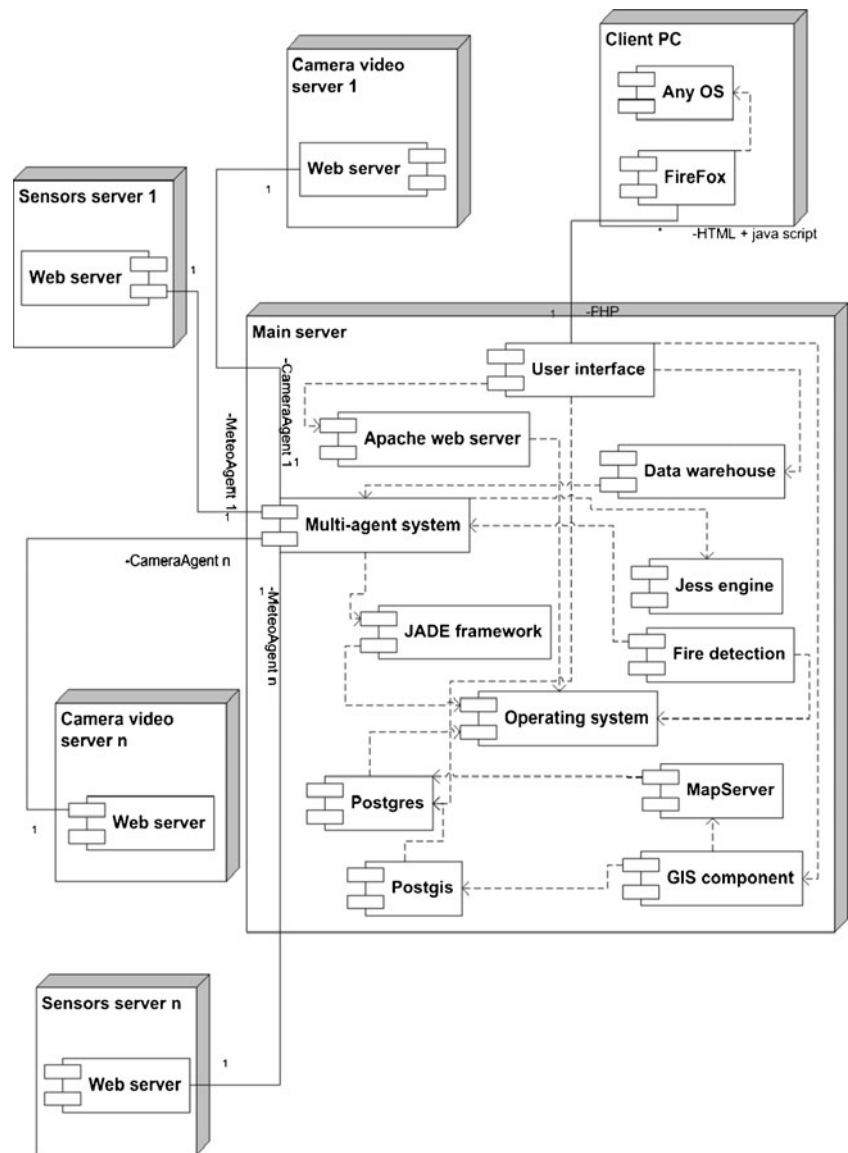
Monitoring geographically large area requires installing several monitoring stations. Each monitoring station is equipped with several sensors among which are usually

pan/tilt/zoom-controlled video camera and mini meteorological station. In our work, sensors are IP-based and thus all data are accessible via TCP/IP protocol. Typical monitoring location is shown in Figs. 6 and 7.

Each monitoring station provides large amount of data. Meteorological station measures important meteorological parameters like air temperature, relative humidity, air pressure, and wind speed and direction. Additional meteorological parameters like insolation, precipitation, moisture, ground temperature, and ground humidity can also be measured. Additionally, temperature inside the equipment box, lighting strokes and accumulator voltage, and current for autonomous power supply are also measured and used to enhance safe work of the monitoring station. Digital image provided by the video camera is data with the most significance in fire recognition, and each pan/tilt/zoom camera can be treated like up to 16 cameras with predefined preset positions covering the whole surrounding area. All data are required by central processing unit for the fire recognition, and are stored in the database for future analysis.

Gathering the distributed real-time data from several locations is challenging, and our solution is a multiagent system. Intelligent agent, in the context we are using it, is a software entity that acts autonomously, has its own internal knowledge and goals, is aware of its state and surroundings, and can cooperate with others (agents and other software components). Multiagent system consists of several agent types. Each agent type can be executed in several instances concurrently to do the work. The internal structure of the system is complex. A single monitoring station is controlled with up to 20 agents depending on the monitoring station properties like number of cameras, number of preset positions per camera, ... Agents rely heavily on interoperation between instances of different agent types. The multiagent system architecture is layered with agents on the top level controlling the agents on the lower level (Fig. 8).

Fig. 4 iForestfire UML deployment diagram



Short description of agent types and functionality is provided. Meteo agent with subagents, *Tini meteo* and *Axis meteo*, is designed with the task of collecting data from each sensor in the sensor network and storing it into database. These agents require knowledge about a location and a protocol for acquiring data provided by the Database agent. Camera agent with subagents, *Preset agent 1, ..., n* (depending on the number of preset positions), is designed with the task of collecting image from video camera and storing it into data warehouse. These agents also require knowledge, provided by the Database agent, about a location and a protocol for acquiring an image.

Multiagent system architecture is highly modular. If monitoring station is not equipped with a video camera, agents dedicated to collecting image from video camera and storing it into data warehouse are not activated. If monitoring station is not equipped with a meteorological

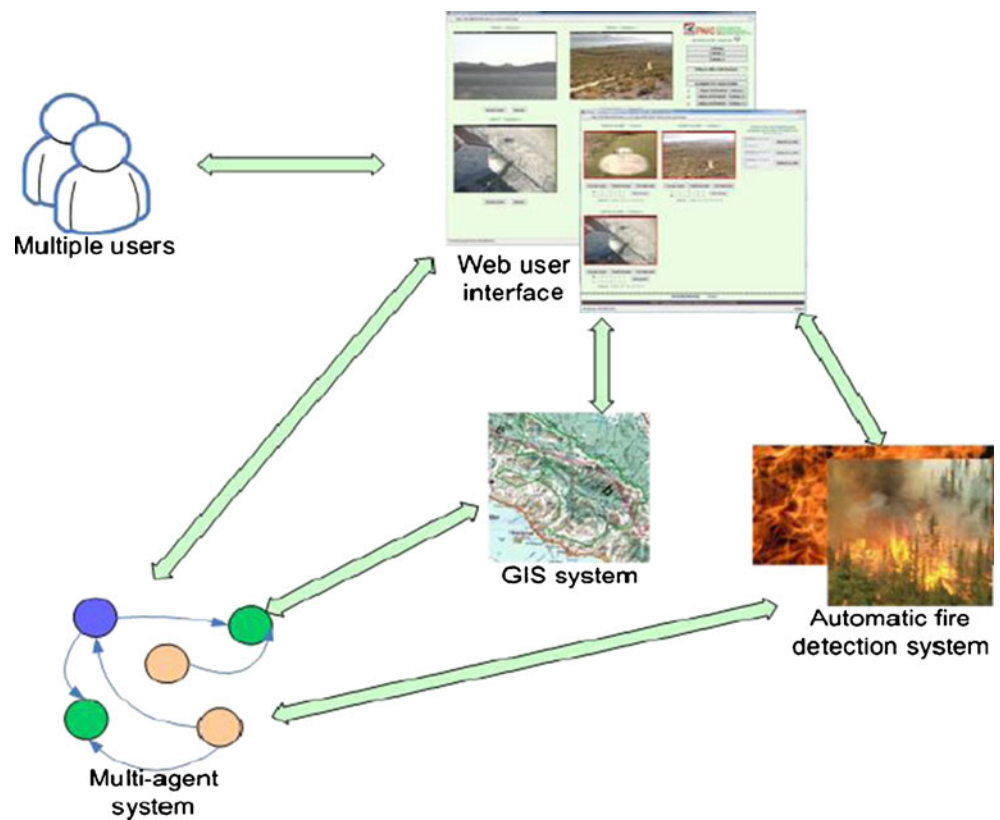
station, agents dedicated to collecting meteo data from each sensor of sensory network and storing it into database are not activated. System is implemented using JADE (Bellifemine et al. 2007; Han et al. 2010) (Java Agent Development Environment) and running Rete (Forgy 1982) algorithm for reasoning and knowledge processing.

3.2 Automatic fire detection

Early recognition of forest fire and appropriate fast reaction is the only way to minimize damage and threat to the infrastructure and human lives. The central part of the iForestFire is the component for the automatic detection of the fire. The detection is based on the visible signs of forest fire, as well as on the inputs from other sensors.

The fire flame in its early stages is usually not visible in daylight, especially if the monitoring spot is far from the

Fig. 5 iForestfire high level functional software components



location of the fire. However, the visible feature of the forest fire in densely wooded areas that can be used to detect fire in its incipient stage is a smoke. Visible signs of fire are much easily recognized during the night, when the fire produces high contrast to the unenlightened landscape. The automatic detection system has two operational modes for the detection

of the fire in daylight and during the night. The mode of selection is automatic based on the video input.

Regardless of the operational mode, detection is carried out in up to 16 preset positions covering the entire field of

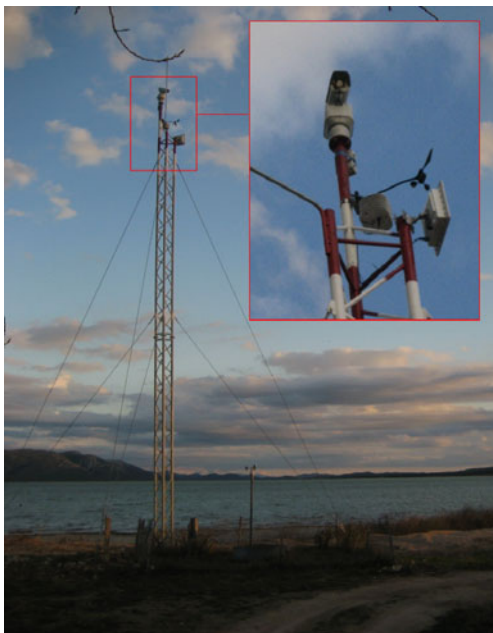
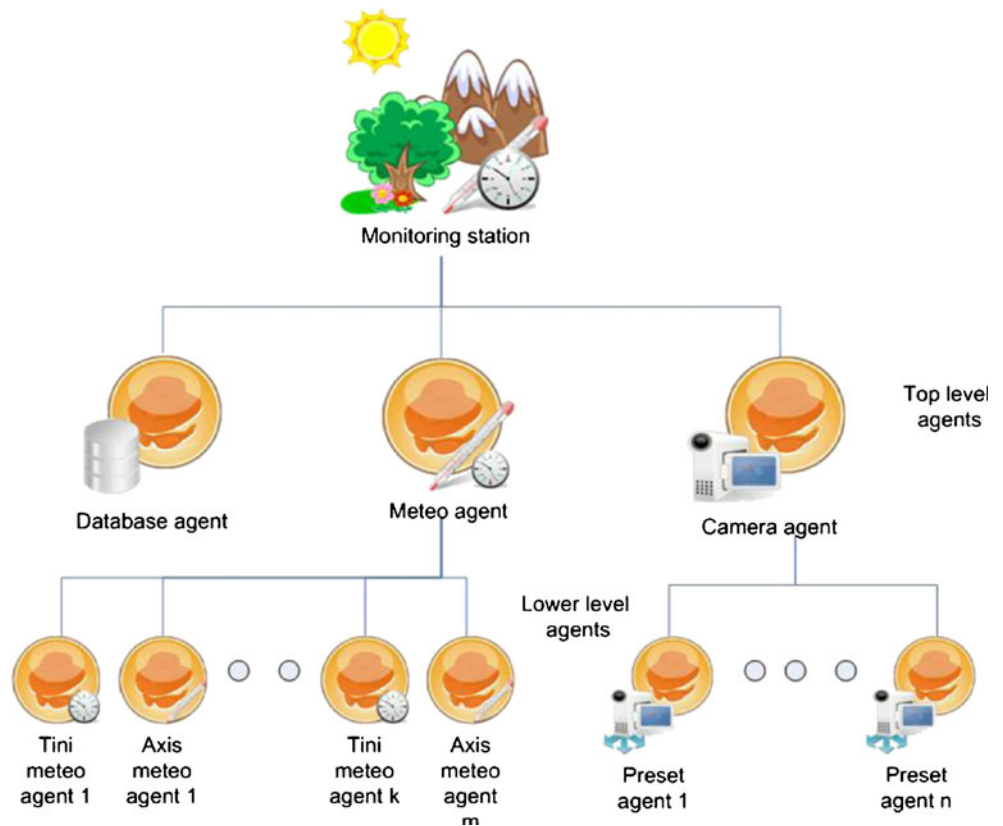


Fig. 6 Stand-alone monitoring station



Fig. 7 Monitoring station mounted on existing infrastructure (astronomical observatory)

Fig. 8 iForestfire multiagent system architecture



view of the camera. The detection in single preset position takes about 15 s, resulting in up to 4 min interval between two detection cycles.

If the suspicious region in the visual range of the camera is detected, the alarm is delivered to the human operator who makes the final decision about delivering the alarm to the fire fighters or discarding a false alarm. The sensitivity of the automatic fire detection can be adjusted using several parameters; thus the system can be easily tuned for different landscapes and particular atmospheric and illumination conditions.

To detect smoke with reasonably low error rates, several algorithms based on different visual characteristics of smoke are implemented. Postprocessing algorithms based on meteorological and video data fusion are applied and decision about raising an alarm is brought by a voting-based strategy where weight is assigned to the output of each detection algorithm.

The first processing step in smoke detection is motion detection and background subtraction. Several methods have been proposed in the literature (Benezeth et al. 2008; Piccardi 2004). Time and space complexity constraints enforced by simultaneous processing of several video inputs in real time impose the selection of an algorithm with low computational complexity and memory requirements. Accordingly, moving pixels detection is based on

the background subtraction method proposed in Collins et al. (Collins et al. 2000). A foreground blob of pixels b_n at time step n is defined by

$$b_n = \{x : |I_n(x) - B_n(x)| > T_n(x)\} \tag{1}$$

where $B_n(x)$ is the background, $T_n(x)$ is threshold, and $I_n(x)$ is the current frame value at the pixel x and time step n . Both the background B_n and the adaptive threshold T_n are recursively estimated from the sequence of frames I_0, \dots, I_{n-1} :

$$B_n = \begin{cases} \alpha B_{n-1}(x) + (1 - \alpha)I_n(x), & x \in b_n \\ B_{n-1}(x), & x \notin b_n \end{cases} \tag{2}$$

$$T_n = \begin{cases} \alpha T_{n-1}(x) + 5 \times (1 - \alpha)|I_n(x) - B_{n-1}(x)|, & x \in b_n \\ T_{n-1}(x), & x \notin b_n \end{cases} \tag{3}$$

where α is a time constant that specifies how fast new frames are adopted in the background model. Initial background B_0 is taken to be the first frame in sequence and threshold T_0 is set to a predefined value.

We have adapted this method to the multiple position detection system with a long time interval between two

visits to the same position. To cope with the long pause between two subsequent frames, at the beginning of each detection cycle in the particular position, background model is updated with the first frame in the new cycle:

$$B_n(x) = \delta B_{n-1}(x) + (1 - \delta) I_n(x), \quad (4)$$

$$\frac{2}{3} \alpha < \delta \leq \alpha$$

The above equation takes into account the changes that have occurred as a result of different illumination or environmental conditions between two visits to the same preset position. Constant δ defines the influence of first frame in sequence to the previously adopted background model. If a new object enters the scene between two visits to the same position, it will still be detected. Even if the larger change is erroneously adapted to the background model, dynamic objects, like smoke, will be detected in the subsequent frames.

Suspicious regions disclosed by background subtraction are further processed by several algorithms based on different visual characteristics of the smoke. As the system is designed for monitoring wide areas, smoke can be detected several miles from the camera position, and thus the texture information content is usually low. However, color, histogram features and shape attributes, both intra-frame and temporal can be used to distinguish smoke-like clusters of pixels from other artifacts in the input video stream.

Color and histogram characteristics are empirically acquired from the image collection gathered on the real forest-fire monitoring sites (Bodrozic et al. 2006) and images from the archive of the Professional Firefighting Brigade of the Split-Dalmatian county of the Republic of Croatia. Pixel-level segmentation of smoke colored pixels (Krstinic et al. 2009) incorporates probabilistic model to classify a pixel into the *Smoke* class (ω_s) or into the *Non-Smoke* class (ω_{ns}). Pixels belonging to the *Smoke* class are assumed to have measurement vector x (color coordinates in HSI color space) distributed according to the distribution density function $p(x|\omega_s)$ and the distribution of the Non-Smoke class is defined with $p(x|\omega_{ns})$. Once the distributions have been estimated, the Bayes theorem is applied to calculate the probabilities:

$$p(\omega_s|x) = \frac{p(x|\omega_s)p(\omega_s)}{p(x)} \quad (5)$$

$$p(\omega_{ns}|x) = \frac{p(x|\omega_{ns})p(\omega_{ns})}{p(x)} \quad (6)$$

where the prior probabilities $p(\omega_s)$ and $p(x|\omega_{ns})$ represent the probabilities of *Smoke* and *Non-Smoke* classes before

observing the vector x . The newly encountered pixel, represented with the measurement vector x , is classified as smoke if

$$\frac{p(\omega_s|x)}{p(\omega_{ns}|x)} > 1 \quad (7)$$

Prior probabilities $p(\omega_s)$ and $p(\omega_{ns})$ can be estimated from the training data: if a random sample of the entire population has been drawn, the maximum likelihood of $p(\omega_s)$ is just the frequency with which ω_s occurs in the training data set. In practice, real forest fires are very rare on any monitoring site, which would result in $p(\omega_s) \ll p(\omega_{ns})$. We use the prior probabilities as a user-controllable parameter, which controls the sensitivity of the smoke detection algorithm. This way the algorithm can be biased to minimize more expensive errors (it is obviously more serious to miss the detection of a real forest fire than to disturb the operator with a false alarm). Probability distributions for the *Smoke* and *Non-Smoke* classes are computed from the training set of images using the kernel density estimation technique based on the assumption that the probability distribution at a continuity point can be estimated using the sample observation that falls within region around that point (Fukunaga 1993; Scott and Sain 2004; Meer 2004).

Each input data sample (image pixel) in the training data set is assigned a kernel, decreasing monotonically with the distance from the origin. Density at some point x is computed as the sum of the contributions of all data samples:

$$f^D(x) = \frac{1}{Nh^d} \sum_{i=1}^N K \frac{x - x_i}{h} \quad (8)$$

where N is the number of samples, h is the bandwidth, and kernel $K: \mathbb{R}^d \rightarrow \mathbb{R}$, $K(x) \geq 0$ is a symmetric function satisfying

$$\int_{\mathbb{R}^d} K(x) dx = 1 \quad (9)$$

In the discrete histogram, pixel contribution is accounted for in a set of cells surrounding pixel origin in the targeted color space, rather than only in one cell. This approach compensates the error introduced by the discretization of the feature space (Krstinic et al. 2009), resulting in the probability distribution that better reflects the underlying true distribution.

In addition to pixel-level color segmentation of smoke colored pixels, histogram features, based on the histogram of a region of the image are also used to distinguish smoke-like objects from other disturbances isolated by background

subtraction. Dispersion $\hat{\mu}_1$ and mean m_1 are computed as (Jain 1989, p 334):

$$m_1 = \frac{1}{N} \sum_{i=1}^N x_i \tag{10}$$

$$\hat{\mu}_1 = \frac{1}{N} \sum_{i=1}^N |x_i - m_1| \tag{11}$$

for the *intensity* and *saturation* components of the suspicious regions on the input image in *HSI* color space. Obtained values are compared with the ranges empirically acquired from the training data set.

Background subtraction and confirmation based on color and local histogram can efficiently identify the appearance of smoke in the controlled region with zero miss rate. However, there are situations when the smoke does not exist in reality, but the automatic surveillance system recognizes characteristics typical to the phenomena, which leads to an alarm that does not correspond to the real occurrence of forest fire. Primary causes for false alarms are natural phenomena visually similar to smoke that at certain conditions can be misinterpreted as smoke even by a human observer. These include fog and clouds that are low to the ground (Fig. 9), dust from the ground, etc. Other false detections include those that can be easily dismissed by common-sense reasoning of the human observer, like rain drops on the camera (Fig. 10), intense changes in environmental illumination, and sun effects, etc. In fact, the largest number of false alarms occurs in periods of the dynamic changes in the environmental illumination, especially in dawn and dusk. Such dynamic changes can be easily detected as motion, which in turn triggers the process of detection.

Different scenarios that can trigger false alarm have to be covered by specific methods. Thus, several methods have been developed to reduce the false alarm rate (Jakovcevic et al. 2009). These methods are mainly based on spatial characteristics of objects segmented as possible smoke formations. Spatial attributes can be divided into intra-frame attributes and temporal characteristics. The latter are mainly used to eliminate abrupt changes detected as smoke. Though the appearance of smoke is gradual, dynamic changes in aerial illumination can result in large regions of image detected as smoke (Fig. 11). Accordingly, if the ratio of object size vs. evolution time is above a certain threshold, the segmented object is discarded.

Intra-frame attributes are based on the geometry of suspicious objects detected on a single frame and on the observation that the smoke plumes have rather convex shape that is not over-elongated. For all detected objects, the axis of least moment of inertia is calculated,



Fig. 9 Low clouds and fog can easily be misinterpreted as smoke

corresponding to the intuitive length of the object, which is used to calculate elongation factor (Jakovcevic et al. 2009), ratio of length vs. width of the object. Experimental values acquired from the training set of images showed that the smoke in the incipient stage of forest fire have elongation factor less than 3. Elongation factor is efficiently used to reject majority of alarms triggered by sunlight effects on camera objective (Fig. 12).

False alarms triggered by raindrops and filth on the camera objective can be rejected based on the observation that smoke has no compact shape (Fig. 13), whereas objects like raindrops are rather circular and compact (Fig. 14). Compactness factor (Jakovcevic et al. 2009) is calculated using the perimeter of the object and its area:

$$c = \frac{l^2}{4\pi A} \tag{12}$$

where l is the perimeter of the object and A is the area of the object. Another feature that can be used for distinguishing



Fig. 10 False alarm caused by rain drop on the camera objective



Fig. 11 False alarm triggered by abrupt change in environmental illumination (have large size vs. evolution time ratio)

smoke from raindrops is the curvature of the smoke shape, which by nature of its spread is rather distorted. Bending energy of the object shape is calculated according to:

$$B = \frac{1}{l} \sum \alpha^2 \tag{13}$$

where l is the perimeter of the object and α is the angle between two pixels whose distance is 3 from neighboring boundary pixels (Fig. 15).

Besides methods based solely on image processing and analysis techniques, alarms can also be discarded using more complex approaches based on sensor data fusion in combination with common sense reasoning. These methods combine information extracted from video input with meteorological information. Thus, phenomena like raindrops and fog can be recognized by using a moisture detector. Temporal characteristics of smoke could be analyzed by comparing optical flow in suspicious regions



Fig. 12 False alarm triggered by sunlight effect on the camera objective (easily rejected using elongation factor)

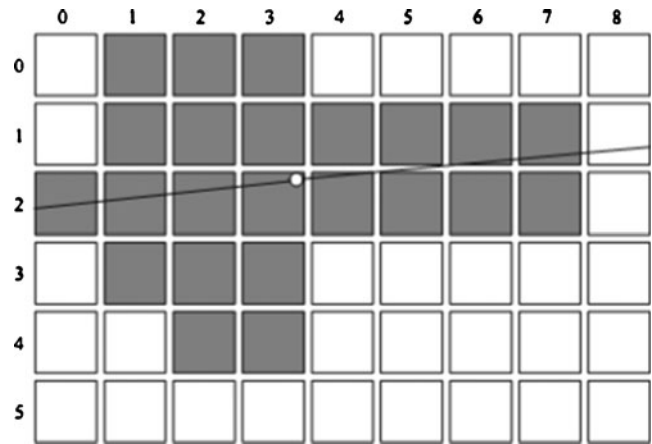


Fig. 13 Axis of least moment of inertia

in image with the wind direction and speed (Fig. 16). However, extreme importance of keeping zero miss rate in fire recognition suggests that no alarm should be dismissed based solely on meteorological information, because they reflect meteorological situation on micro location where the equipment is mounted, whereas fire and smoke can be recognized several miles from the monitoring spot.

Besides the above-mentioned methods used for automatic forest fire detection, several different approaches were tested and evaluated, and the algorithm is continuously upgraded. However, severity of the problem enforces constraints for the implementation of detection methods in the real system. First is the system reliability, which enforces constraints in the complexity of the used methods with regard to the stability of the system and computational resources needed for simultaneous handling of several video inputs.

Different problems arise from the versatile conditions in which the system operates. Variations in the smoke color tones, environmental illumination, atmospheric conditions (clouds, mist, shadows, dust), varying quality of images of wide outdoor areas, filth on the camera lens installed on difficult accessible sites, and other problems make smoke detection a complex task. If a method is to be implemented as part of the detection algorithm, it should produce detection with reasonably low error rates in all possible conditions.

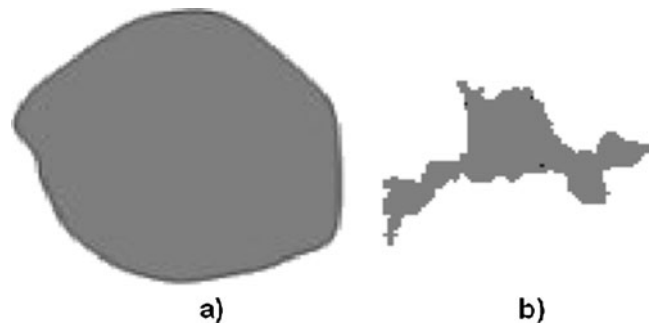


Fig. 14 Detected objects a) raindrop, b) smoke

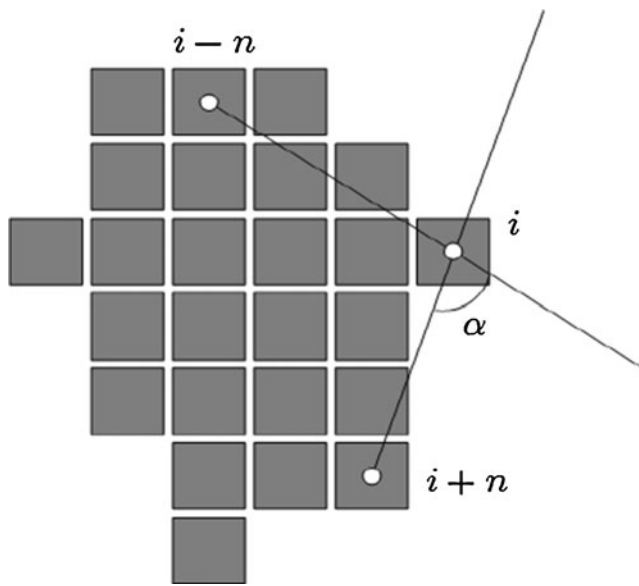


Fig. 15 Angle α calculation

All methods can be roughly divided into intra frame (based solely on current image) and inter-frame. Inter-frame methods explore temporal characteristics and changes in the sequence of the images. Basic inter-frame method is the motion detection and background subtraction, which is used as the first processing step of the detection algorithm. Deeper insight and information on temporal characteristics of the regions in the visual field of the system can be obtained by Markov model analysis and time-domain wavelet transformation. However, regions that are far from the camera do not reflect the actual dynamic characteristics of the phenomena observed; thus temporal characteristics of the smoke cannot be reliably detected with these methods. Further, smoke can appear several miles from the monitoring station as well as at a few hundred meters from the camera. Besides dynamic characteristics of the smoke that rapidly changes depending on the distance from the camera,



Fig. 16 Optical flow detection

intra-frame attributes are also mutable. This makes methods like frequency-domain wavelet analysis and neural networks, which heavily rely on texture information and intra-pixels relations, hard to implement and unreliable.

The above-mentioned approaches were tested first in house conditions and then implemented in the experimental system mounted Zvezdano selo research center. On the basis of thorough analysis, methods that satisfy stability and speed requirements are implemented as part of the production system.

3.3 Data archiving and retrieval

Data are archived during the process of data gathering from video and meteo sensors in real time. Agents store data in a database and data warehouse (images). The collected data are accompanied with information relevant to the data retrieval, like location from which data are acquired and date when data are acquired. This metadata enables the system to provide the correlated data to the user. For example, correlated data enable retrieval of images acquired directly from a camera and images generated during the image processing from those “original” images, the so-called alarm images. Data retrieval always starts from the database. All metadata information is organized in the database. Image data retrieval is expanded from the database to the data warehouse. The storage is limited with the server hardware. The user can choose different hardware. A server can have 40 or 200 Gbytes of permanent storage. The iForestFire system can be administrated to work regardless of the server storage changing the system setup. Data stored in the database are small and do not present a problem, but archived images in the data warehouse require larger capacity. Usually, images are collected as colored images with 768×576 pixel dimension. Depending on the camera quality, each image is approximately ~ 50 Kbyte big. Collected images from video cameras can be set up with different image properties. For each camera, image is stored for every preset position defined on the camera. If the system has two cameras with eight preset positions and camera makes one full circle per minute, collected image data is ~ 800 Kbytes/min. Storage would be filled soon, so algorithm for cycle overwrite of images is defined. Data archiving and retrieval component uses the algorithm for image archiving. Image identifier identifies the image uniquely in the data warehouse. It includes image number from 0 to the image limit number. Image limit number is a part of the system setup and can be administrated by the user. Usually, image limit number is set to 10,000. When that number of images is reached, the counter is reset back to 0 and acquired images overwrite the already stored images with the same image identifier. With these settings, about one month’s input images and one year’s alarm images can be stored for later analysis to 60 Gbytes of

permanent storage. Our experience gained in commercial implemented systems shows that this is enough.

Metadata organization for archived images is shown in Fig. 17.

Metadata for each archived image contains information of camera location from which image is acquired, preset position of the camera when image is collected, date and time when image is taken, unique image identifier of the image in the data warehouse, and information whether the image is original or an alarm image generated from original image by the automatic fire detection component. User, wanting to retrieve archived images from iForestFire system, can use image metadata to retrieve images and to limit retrieved images by camera location, preset position, date and time and original/alarm image. Figure 18 shows the web page that provides the user with the selection of parameters for image retrieval with retrieved images presented to the user in a form of sliding sequential images.

Meteorological data and metadata on meteorological data are stored in the database. Metadata for each archived meteorological data is accompanied with information on meteorological station location from which the data are acquired, type of meteorological data (e.g. temperature, wind speed, etc.), and date and time the data are taken. A user wanting to retrieve archived meteorological data from iForestFire system can use metadata to retrieve meteorological data and to limit retrieved data by a location, date and time, and a type of meteorological data.

Figure 19 shows the web page that provides a user with the selection of parameters for meteorological data retrieval with retrieved data presented to the user in the form of a graph with selected type of meteorological data drawn

Fig. 17 Image metadata

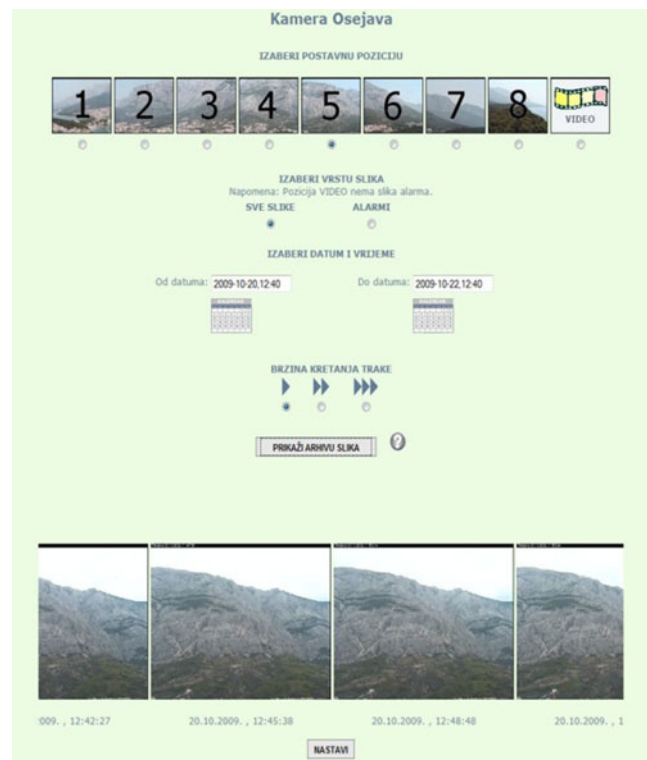
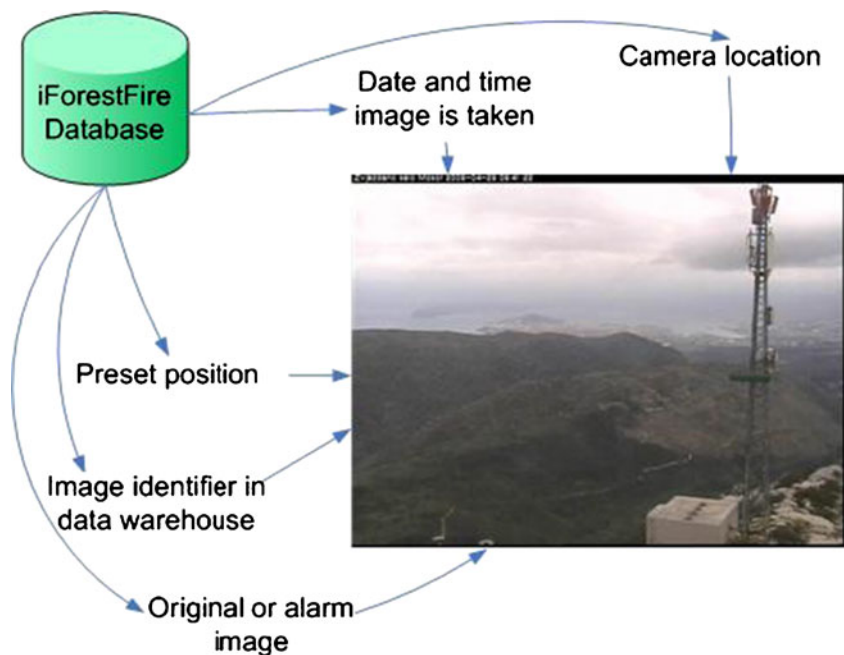


Fig. 18 User interface for archived image retrieval

together for comparison. In this figure, only wind speed and temperature are selected for retrieval.

Meteorological data storage is limited with the database setup and hardware capacity. There are installed iForestFire system (National Park Paklenica, Nature Park Vransko jezero, Nature Park Biokovo all in Croatia) running for 2 years continuously without reaching capacity limit.

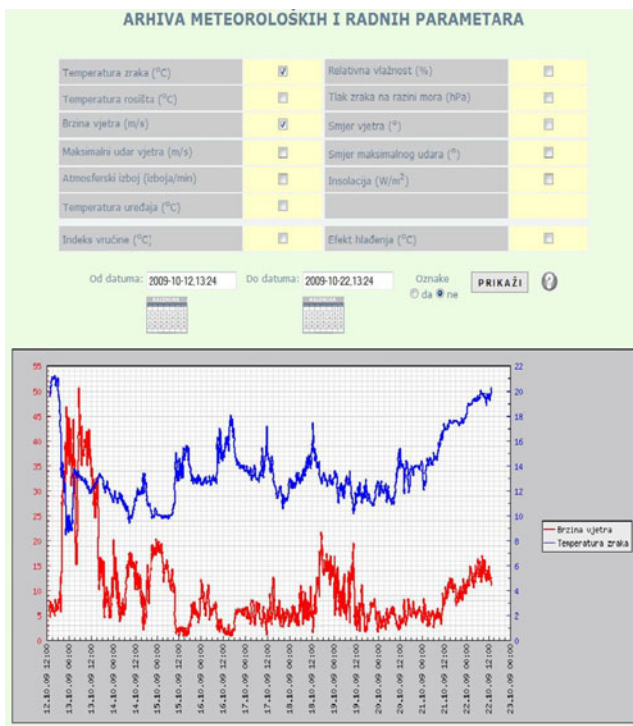


Fig. 19 User interface for archived meteorological data retrieval

3.4 Geolocation information system for positioning sensors in space and fire risk and spreading simulation

As MacEachren (MacEachren 2010) said location is fundamental to crisis management. iForestFire integrates GIS information into several system functions. Geolocation information system is used to determine the position of monitoring stations. The user can locate monitoring station in space using GIS map in the iForestFire. That functionality is provided by georeferencing monitor station’s position (Stipanicev et al. 2009a). This component also enables the user to direct video surveillance from multiple monitoring stations in one direction by simply clicking on the area on the georeferenced map.

GIS is important and used in all modes, but in simulation mode, it is essential for fire behavior simulation and fire spread calculation. Fire risk and spreading simulation component (called MOPP-modeling of fire propagation in Croatian) is based on semi-empirical Frandsen-Rothermels fire spreading model (Rothermel 1972) and raster model based on cellular automata for two-dimensional fire spreading simulation. Open space vegetation can be observed as two-dimensional cellular automata. Each cell presents attributes according to environment properties, first of all the vegetation flammability properties and terrain topography.

These two components provide a user with:

- web interface for GIS information on fire initial location,

- calculation of fire spreading,
- web visual simulation of fire spreading overlaid with GIS data for the current meteorological data
- web visual simulation of fire spreading overlaid with GIS data with user-set meteorological data for the if-then situation (if the wind speeds up then how will the fire spread?)

MOPP component is implemented on the top of open source GRASS service for GIS calculation and MapServer for presentation and GIS layers manipulation.

3.5 Comparison of iForestFire with similar systems

iForestFire can be compared with similar commercial available systems. Automatic fire detection systems are primarily divided into satellite-based and terrestrial-based systems. Representatives of satellite-based systems are Canadian FireM3 (Fire Monitoring, Mapping, and Modeling 2010), European EUMETSAT FIR (EUMETSAT Active Fire Monitoring 2010), and NASA MODIS (MODIS Rapid Response System 2010). Although satellite image can cover large area, the main disadvantages of satellite-based systems are low temporal and spatial resolution of images. Terrestrial-based systems are based on terrestrial monitoring stations with different sensors (infra-red camera, visible spectra camera, lasers, optical spectrometer, laser, temperature sensor, etc.).

Representatives of terrestrial-based systems are German system FireWatch (FireWatch 2010) that uses special video detector and Portuguese ForestFireFinder (Forest Fire Finder 2010) that uses optical spectrometry. These systems are fire detection systems, and when compared with iForestFire, these systems lack tele-presence for fire fighters. As stated in Introduction, our experience (Stipanicev et al. 2006) in system development through users’ feedback has shown that fire fighters intensively use tele-presence and manual control of video cameras. Another advantage of iForestFire when compared with similar systems is that the system can work with different equipment (Samsung, Sony, Pelco cameras,...). The only requirement for video cameras and other sensing devices is accessibility via http (Hyper-Text Transfer Protocol). The existing infrastructure in Istra county was installed by fire fighters for monitoring and was upgraded to automatic fire detection with iForestFire with very little adjustment. iForestFire is quite open to different hardware and software solutions, enabling integration of different COTS (commercial off-the-shelf) components. Openness of the system enables reuse of the existing monitoring equipment and upgrading the existing plain monitoring systems to systems with monitoring and automatic fire detection. Monitoring and fire detection

integration is used in other similar systems like South African FireHawk, French UraFire, and others. iForestFire is a web-based information system. This enables access to system from anywhere. It also facilitates publication of gathered information to a general public in the form of environment hazards notification systems.

4 Conclusion

Intelligent environmental sensor networks provide a way for monitoring and controlling the physical environments. Information collected by these systems gives new insights into the environment and provides data for modeling and simulation of environmental processes. Intelligent processing of the collected data in real time can reduce hazards of natural and man-made disasters by detection of dangerous situations and raising early warnings. These systems rely on the sensor and communication technologies to collect and transmit data from field sensors, data warehousing technology, and computational and artificial intelligence algorithms used to generate complex conclusions and decisions.

Such a system is Intelligent Forest Fire Monitoring System presented in the paper. Croatia as well as whole Mediterranean area belongs to regions with a high forest fire risk. In summer season, seven coastal counties in Croatia and in particular the Adriatic islands are permanently exposed from high to very high fire risks, due to densely spaced conifer forests. The only effective way to minimize damage caused by forest fires is their early detection and appropriate fast reaction. The presented system is based on a network of remotely controlled video cameras and meteorological stations integrated with the geolocation information system and intelligent data processing algorithms. Collected data are processed in real time to provide early detection of possible forest fire. In the case of a real incident, video presence is used to guide firefighters' efforts on the terrain and to prevent dangerous situations, whereas the fire risk and spreading simulation module is used to anticipate development of the fire.

iForestFire system benefits from multiagent technology for data gathering and data integration in a highly modular concept of the system. Agents are particularly suitable to improve the system robustness and decrease system centralization while the system becomes more open and easily expandable by simply adding new agents.

The system is developed in tight cooperation with users, primarily fire fighters, and improved to the current state through several iterations. Tight involvement of users during entire SDLC ensures that iForestFire is highly user-tailored. Also, openness to different hardware solutions facilitates reuse of the existing monitoring

equipment and upgrading it to more automatized environment protection.

Intelligent Forest Fire Monitoring system presented in this paper is installed on several locations on the Croatian coast and islands. Currently, the best coverage is achieved in Istra county where complete coverage of the peninsula of Istra is achieved by 29 field units (cameras and mini meteorological stations) connected to seven data processing centers. Systems installed on different locations have detected several real forest fire incidents. Video archive of the systems has also been used in police investigations about incidents.

The presented system is highly modular and different modules can be used separately or implemented as a part of similar monitoring systems. Thus, by implementing alternative algorithms in automatic mode, the system can be easily adapted for monitoring and detection of other natural phenomena.

References

- Beck, K., & Andres, C. (2005). *Extreme programming explained: Embrace change (2nd Edition)*. Addison-Wesley.
- Bellifemine, F. L., Caire, G., & Greenwood, D. (2007). *Developing multi-agent systems with JADE*. John Wiley & Sons.
- Benezeth, Y., Jodoin, P. M., Emile, B., Laurent, H., & Rosenberger, C. (2008). Review and evaluation of commonly-implemented background subtraction algorithms. *Proc. 19th International Conference on Pattern Recognition*, Tampa, Florida, USA, pp 1–4.
- Bodrozic, L. J., Stipanicev, D., & Stula, M. (2006). Agent based data collecting in a forest fire monitoring system. *Proc. International Conference on Software, Telecommunications and Computer Networks*, Split-Dubrovnik, Croatia, pp 326–330.
- Collins, R., Lipton, A., Kanade, T., Fujiyoshi, H., Duggins, D., Tsin, Y., et al. (2000). A system for video surveillance and monitoring. *Tech. report CMU-RI-TR-00-12*, Robotics Institute, Carnegie Mellon University.
- EUMETSAT Active Fire Monitoring (FIR). <http://www.eumetsat.int/>. Accessed 07 December 2010.
- Fire Monitoring, Mapping, and Modeling (Fire M3). http://cwfis.cfs.nrcan.gc.ca/en_CA/background/summary/fm3. Accessed 12 October 2010.
- FireWatch. <http://www.fire-watch.de/cms/>. Accessed 29 September 2010.
- Forest Fire Finder. http://www.ngns-is.com/html/florestas/fff_intro_eng.html. Accessed 29 September 2010.
- Forgy, C. (1982). Rete: a fast algorithm for the many pattern/many object pattern match problem. *Artificial Intelligence*, 19, 17–37.
- Fukunaga, K. (1993). Statistical pattern recognition. In C. H. Chen, L. F. Pau, & P. S. P. Wang (Eds.), *Handbook of pattern recognition and computer vision* (pp. 33–60). River Edge: World Scientific Publishing Co.
- Han, S., Youn, H. Y., & Song, O. (2010). Efficient category-based service discovery on multi-agent platform. *Information Systems Frontiers*, doi:10.1007/s10796-010-9281-x
- Isakowitz, T., Bieber, M., & Vitali, F. (1998). Web information systems. *Communications of the ACM*, doi:10.1145/278476.278490

- Jain, A. K. (1989). *Fundamentals of digital image processing*. Prentice-Hall, Inc.
- Jakovcevic, T., Stipanicev, D., & Krstinic, D. (2009). False Alarm Reduction in Forest fire Video Monitoring System. *Proc. of MIPRO 2009, 32nd International Convention, Opatija, Croatia, 3*, 265–270.
- Krstinic, D., Stipanicev, D., & Jakovcevic, T. (2009). Histogram-based smoke segmentation in forest fire detection system. *Information Technology and Control, 38*(3), 237–244.
- MacEachren, A. M. (2010). Geovisual Analytics for Crisis Management: Moving Beyond GIS. *ISCRAM'10 Conference Proceeding 2010*, Seattle, USA.
- Meer, P. (2004). *Robust techniques for computer vision*. In G. Medioni & S. B. Kang (Eds.), *Emerging topics in computer vision* (pp. 107–190). Prentice Hall.
- Ministry of the Interior, statistical report. http://policija.hr/mup.hr/UserDocsImages/statistika/2007/statistika_pozari.pdf. Accessed 20 September 2010.
- MODIS Rapid Response System. <http://rapidfire.sci.gsfc.nasa.gov/>. Accessed 10 July 2010.
- Piccardi, M. (2004). Background subtraction techniques: a review. *Proc. of IEEE SMC 2004 International Conference on Systems, Man and Cybernetics, 4*, 3099–3104.
- Puras, J. C., & Iglesias, C. A. (2009). Disasters2.0. Application of Web2.0 technologies in emergency situations. *Conference on Information Systems for Crisis Response and Management, ISCRAM'09 Conference Proceeding 2009*, Gothenburg, Sweden.
- Rothermel, R. C. (1972). A mathematical model for predicting fire spread in wildland fuels, *Research Paper INT-115*. United States Department of Agriculture, Forest Service, Intermountain Forest and Range Experiment Station, Ogden.
- Scott, D. W., & Sain, S. R. (2004). *Multi-dimensional density estimation*. In C. R. Rao & E. J. Wegman (Eds.), *Handbook of statistics—data mining and computational statistics* (v.23, pp. 229–263). Elsevier.
- Seric, L. J. (2010). *PhD report Agent based environmental intelligence*, Faculty of Electrical Engineering, Mechanical Engineering and Naval Architecture, University of Split.
- Stipanicev, D., Vuko T., Krstinic, D., Stula, M., & Bodrozic, L. J. (2006). Forest fire protection by advanced video detection system—croatian experiences. *Proceedings of the third TIEMS Workshop-Improvement of Disaster Management System*, Trogir, Croatia.
- Stipanicev, D., Bugarcic, M., & Seric, L. J. (2009a). Integration of forest fire video monitoring system and geographic information system. *Proc. of the 51st Int. Symp. ELMAR 2009*, Zadar, Croatia, pp 1–5.
- Stipanicev, D., Stula, M., Krstinic, D., & Seric, L. J. (2009b). Agent based intelligent forest fire monitoring system. *Proc. of MIPRO 2009, 32nd International Convention, Opatija, Croatia, 3*, pp 260–265.
- Summary Report on the Status of Environmental Monitoring and Reporting in Europe by Nesis project. http://www.nesis.eu/documents/d3.1.sop_report_revised%20version_v6.1_100331.pdf. Accessed 01 October 2010.
- Szyperski, C. (1998). *Component software: beyond object-oriented programming*. Addison-Wesley.

Maja Stula is associated professor of computer science at the Faculty of Electrical Engineering, Mechanical Engineering and Naval Architecture, University of Split. She received a BS in electrical engineering in 1996, a MSc in 2001 and a PhD in 2005. all from the Faculty of Electrical Engineering, Mechanical Engineering and Naval Architecture, University of Split. Her research interests include intelligent technologies like multi-agent systems, ontology and fuzzy cognitive maps, especially engineering applications of intelligent technologies. She is a coauthor of iForestFire system. Contact information: FESB, R. Boskovic bb, 21000 Split, Croatia; maja.stula@fesb.hr

Damir Krstinic graduated in B.Sc. in 1999. He received a M.Sc. in 2004, and Ph.D. in 2008 from the University of Split. His current research areas are computer vision and artificial intelligence. He participates in the development of the intelligent video based monitoring system for early detection of forest fires, which was awarded as the most innovative Croatian IT industry product for the year 2007 (he is a coauthor of iForestFire system). Other topic of interest are data mining and language processing. He was involved in the development of the web search engine and document clustering algorithm. Damir Krstinić is a senior research assistant at the Faculty of electrical engineering, mechanical engineering and naval architecture of the University of Split

Ljiljana Seric is a junior researcher at Faculty of Electrical Engineering, Mechanical Engineering and Naval Architecture, University of Split, a member of Department for Modelling and Intelligent Systems. She is a coauthor of iForestFire system. Her research interests are focused on data processing and agent technology. She received her BC and PhD in electrical engineering and computer science. Additional information can be found on <http://www.fesb.hr/~ljiljana/>