

# PREDICTIVE MODELING OF THE SPATIOTEMPORAL EVOLUTION OF AN ENVIRONMENTAL HAZARD AND ITS SENSOR NETWORK IMPLEMENTATION

*Dimitris V. Manatakis and Elias S. Manolakos*

Department of Informatics and Telecommunications, University of Athens  
email: {dmanatak, eliasm} @di.uoa.gr

## ABSTRACT

Predicting accurately the spatiotemporal evolution of a diffusive environmental hazard is of paramount importance for its effective containment. We approximate the front line of a hazard with a set of line segments (local front models). We model the progression characteristics of these front segments by appropriately modified 2D Gaussian functions. The modified Gaussian model parameters are adjusted based on the solution of a Kullback-Leibler (KL) divergence minimization problem. The whole scheme can be realized by a wireless sensor network by forming dynamically triplets of cooperating sensor nodes along the path of the hazard. It is shown that the algorithm can track effectively the front characteristics (in terms of direction and speed) even in the presence of faulty sensor nodes.

**Index Terms**— Environmental hazard, predictive modeling, WSNs, Kullback-Leibler divergence.

## 1. INTRODUCTION

Being able to predict with reasonable accuracy the spatiotemporal behavior of an evolving environmental hazard (such as a wildfire, oil slick, etc.) is of great importance to civil authorities since it helps them optimize their response and contain the potential damages. Hazard-specific mechanistic or semi-empirical models are commonly used for this purpose. However, such models usually depend on a large number of parameters that are difficult to estimate and thus quite often fail to make good predictions. To address this limitation, many researchers have proposed decision support system architectures which attempt to integrate simulation-based predictive modeling with real-time field sensing into a closed loop system. The majority of such systems reported so far in the literature rely on remote sensing, where sensor data (e.g. satellite spectral images) are used to periodically calibrate simulation models in real-time in order to minimize their prediction errors. These methods are also known as Dynamic Data Driven Assimilation, a research field that has recently drawn the attention of the scientific community due to its expected high societal impact [1,2].

Unfortunately, in many cases, satellite images, or image data in general, is not available, or may not be appropriate, for detecting a diffusing hazard. In such cases, Wireless Sensor Networks (WSNs), that are becoming a mature state of the art technology due to their rapidly dropping cost, may provide a viable alternative for large-scale environmental monitoring type applications. Recently, a number of WSN-based methods have been proposed for detecting the boundaries of a diffusing hazard [3,4]. Their main objective is to identify at each time step the sensor nodes located closest to the evolving front line. Despite their demonstrated capabilities to delineate the area affected by the hazard these schemes suffer by construction from the severe limitation that their accuracy is proportional to the WSN nodes density (requiring thousands of sensors

per  $1km^2$ ), which renders them impractical even for medium-scale environmental monitoring applications. Finally the inability of these algorithms to provide information about the evolution behavior (direction and speed) of the diffusing hazard renders them inappropriate for decision support based on predictive modeling.

A hazard's front can be approximated as a piecewise linear curve. Each line segment of this curve (local front) can be adequately characterized using a small number of parameters (location of end points, orientation, propagation speed). In this paper we model the spatiotemporal evolution of a local front line segment using a modified 2D Gaussian function. This approach allows us to treat the estimation of local front parameters problem as a statistical model parameters updating problem that we have formulated and solved analytically. We show here how this solution can be implemented by a distributed in-network processing algorithm using small-size dynamically formed clusters (triplets) of cooperating sensor nodes. Processing at each node is relatively "light" and fast since it amounts to implementing closed form algebraic expressions and not some iterative slowly converging optimization procedure.

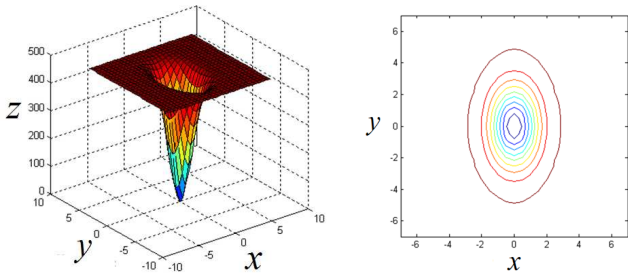
To the best of our knowledge this is the first attempt reported in the literature to use a fully decentralized WSN to implement predictive models, based on signal processing concepts, for the evolution of a hazard's front line. Our work shows how to make decentralized short term phenomenon evolution predictions and forms the basis for developing WSN-supported Dynamic Data Driven Application Systems [1,2] for environmental monitoring and hazard evolution prediction at large-scale.

The rest of the paper is organized as follows: In Section 2 we present the justification for using modified 2D Gaussian functions to model the spatiotemporal evolution of a diffusing phenomenon and present an in-network processing algorithm for estimating their parameters. The Kullback-Leibler divergence minimization problem, which is used as the basis for the local model updating, is formulated in section 3. Experimental validation results are presented and discussed in section 4. Finally our findings are summarized and work in progress is outlined in section 5.

## 2. MODELING AND ALGORITHM DESIGN

### 2.1. Spatiotemporal hazard evolution models

The spatiotemporal evolution of diffusing phenomena initiated from a single point source can be approximated by an evolving ellipsoid with a principal axes ratio depending on area prevailing conditions [5-8] (e.g. wind direction and speed for wildfires, water stream speed and direction for oil slicks etc.). For simulating elliptical spatiotemporal evolution we are introducing the use of a modified 2D Gaussian function whose shape and orientation can be controlled by assigning appropriate values to the covariance matrix elements. Figure 1(a) shows the spatiotemporal evolution (z-axis is time) of a hazard initi-



**Fig. 1.** *a)* Plot of 2D modified Gaussian, *b)* Contour plot of (a)

ated from single point as this is modeled by the proposed function:

$$f_{\theta}(x) = \frac{A}{2\pi|\Sigma|^{\frac{1}{2}}} - \frac{A}{2\pi|\Sigma|^{\frac{1}{2}}} \exp\left(-\frac{(\mathbf{x} - \mu)^T \Sigma^{-1} (\mathbf{x} - \mu)}{2}\right) \quad (1)$$

where the parameters  $\theta = \{A, \mu, \Sigma\}$  are: a predetermined amplitude related constant  $A$ , the mean value  $\mu$  and the 2x2 covariance matrix  $\Sigma = \begin{bmatrix} \sigma_{11}^2 & \sigma_{12}^2 \\ \sigma_{21}^2 & \sigma_{22}^2 \end{bmatrix}$ . In this modeling,  $f_{\theta}(x)$  corresponds to time and the larger the time values (vertical z-axis) the larger the area that is covered by the corresponding iso-temporal ellipsoidal contours on the x-y plain (see Figure 1(b)). The fraction  $\frac{A}{2\pi|\Sigma|^{\frac{1}{2}}}$  corresponds to the largest value that  $f_{\theta}(x)$  can take (modeled time span). The modified 2D Gaussian function (1) has all the properties needed to models spatiotemporal evolution, while also keeping intact all the advantages that 2D Gaussians offer (i.e. adjustable shape, simple parameterization, ability to obtain analytical results).

## 2.2. Sensor network assumptions

Before we proceed with the algorithm's presentation let us state the WSN related assumptions we have made in this work:

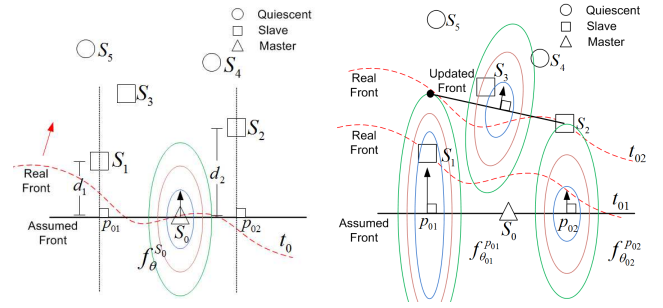
- Sensors are stationary and randomly deployed.
- All sensor nodes have the same processing and communication capabilities.
- Sensor node clocks do not need to be synchronized.
- A sensor  $S$ , can communicate only with its neighbors i.e. the sensor nodes located within the circular communication range of radius  $R$  from its location. It is assumed that each sensor node has at least 2 neighbors.
- Each sensor keeps the following information about itself: {ID, Location, Detection Status Flag, Sensor Status Flag, Prior Model Parameters}
- In addition, each sensor keeps the following information for each one of its neighbors: {ID, Location, Time of (hazard) Detection, Detection Status Flag, Sensor Status Flag}.

## 2.3. The in-network processing algorithm

The main idea of the in-network processing algorithm is as follows: During the evolution of a diffusive phenomenon the deployed sensor nodes are dynamically organized into local clusters of three nodes each. The sensor nodes of a cluster collaborate to update the prior local front evolution belief (model) of the sensor node (Master) that is responsible for forming the cluster. This newly updated model is propagated forward as the phenomenon evolves. Below we provide more details on the implementation of the distributed estimation algorithm.

**Sensor states, self-organization into clusters:** A sensor may assume one of the following states:

**Quiescent:** Default state. The Sensor Status Flag (SSF) has value 0.



**Fig. 2.** *a)* The prior model used by Master  $S_0$  for local front evolution *b)* The estimated models at the intersection points  $p_{01}, p_{02}$  that have minimum KL from the prior; the updated front segment and the updated model  $f_{\theta_{02}}^{p_{02}}$ .

**Master:** It is responsible for the estimation and the update of the local front parameters. The SSF of a Master has value 1.

**Slave:** It is responsible for monitoring the phenomenon upon a request from a Master. The SSF value for Slave sensors is 2. Upon detection of the phenomenon, its Detection Status Flag (DSF) becomes 1 (default value is 0) and is broadcasted to its neighbors. A Slave may serve more than one Masters at any given time.

The presentation of the proposed distributed algorithm will be facilitated by using a simple running example. In Figure 2 we assume w.l.o.g. that the real local front is the red discontinuous curve and its local speed and direction are indicated by the length and direction of the corresponding red vector. In this example we assume (w.l.o.g.) that a sensor is detecting the phenomenon when it is reached by the evolving front.

When the front line reaches sensor  $S_0$  at time  $t_0$ ,  $S_0$  detects the phenomenon, sets its DSF = 1, initializes an internal clock and checks its SSF flag. If SSF = 0 (state = Quiescent) and at least two of its neighbors ( $\{S_1, S_2, S_3\}$  in Figure 2(a)) have not detected the phenomenon yet (having two neighbors with DSF = 0 and SSF = 0 or 2 are the necessary conditions for becoming Master),  $S_0$  changes its SSF to 1 and becomes the Master and will attempt to form a new cluster. Subsequently, the new Master sensor  $S_0$  broadcast a "Master Declaration Message (MDM)"  $\{ID = S_0, SSF = 1\}$  which sets the SSF of its neighbors to 2 (they become its Slaves see Figure 2(a)). Now  $S_0$  waits until it has received two hazard event Detection Messages (DM) from two Slave neighbors. Let's assume that at Master's ( $S_0$ ) internal clock times  $t_{01}$  and  $t_{02}$  ( $t_{02} > t_{01}$  w.l.o.g.),  $S_0$  receives the DMs from  $S_1$  and  $S_2$  respectively. A DM contains only the sending sensor's  $[ID]$ . When the Master ( $S_0$ ) receives a DM it updates its neighborhood table and when it has received two such messages it is ready to start the local front parameters updating procedure (described below).

The Master uses its current (prior) spatiotemporal evolution model (a modified 2D Gaussian) and assumes that the local front at its neighborhood can be approximated by a line segment that is perpendicular to the major axis of its prior model (see Figure 2(a)). The speed of motion and the direction, which are determined from the prior model, are denoted by the vector that is perpendicular to the assumed front. Each point on this front evolves as indicated by the prior elliptical evolution model of the Master. This assumption is well justified and has been extensively used by many researchers [5-7].

**Model updating: Finding the intersection points.** The Master  $S_0$  uses the location of slaves  $S_1$  and  $S_2$  (stored in its neighborhood table) and calculates the two points ( $p_{01}, p_{02}$ ) where these two sensor locations project on the assumed local front line segment. (see Figure 2(a)). For each point on the assumed front segment (and thus for pro-

jection points  $((p_{01}, p_{02})$  as well) it is assumed that the phenomenon is best captured by the Master's prior modified 2D Gaussian model. The Master ( $S_0$ ) computes the distances ( $d_1$  and  $d_2$ ) between sensors  $S_1, S_2$  locations and the corresponding projection points  $(p_{01}, p_{02})$ . Subsequently, Master  $S_0$  estimates two new 2D modified Gaussians  $f_{\theta_{01}}^{p_{01}}$  and  $f_{\theta_{02}}^{p_{02}}$  (see Figure 2(b)) which (i) are centered at the projection points  $((p_{01}, p_{02})$ , (ii) have minimum Kullback-Leibler divergence from its prior model, and (iii) assign at the corresponding slave locations  $S_1$  and  $S_2$  time values equal to  $t_{01}$  and  $t_{02}$  respectively. The direction of the major axes of the two new 2D modified Gaussians are the same as that of the prior model but their rate of spread can be different, as indicated by the size of the two vectors originating at the two projection points  $(p_{01}, p_{02})$  in Figure 2(b). In Section 3 we formulate the optimization problem used to estimate the new model parameters (in closed form). The formulated problem can be solved analytically (solution not shown here due to lack of space).

**Estimating the new local front direction:** Master  $S_0$  checks its table to find out which one of the two used Slaves ( $S_1$  and  $S_2$ ) has detected the phenomenon most recently ( $S_2$  in our example w.l.o.g.). The location of this sensor is one point from where the local front line will pass at time  $t_{02}$  (see green curve in Figure 2(b)). A second point is estimated as follows: Master  $S_0$  calculates using the updated modified Gaussian model  $f_{\theta_{01}}^{p_{01}}$  the point on that model's major axis reached after  $t_{02}$  (marked by a black dot in Figure 2(b)). This point and point  $S_2$  are equi-temporal (both are reached by the front at time  $t_{02}$ ) and thus define an estimated local front line segment at the specific time instance.

**Updating the prior model:** One of the models  $f_{\theta_{01}}^{p_{01}}$  and  $f_{\theta_{02}}^{p_{02}}$  will be used as the new spatiotemporal evolution model for the newly formed local front line segment. Assuming smooth model changes, we select among them the model with the smaller KL-divergence from the prior model of the Master  $f^{S_0}$  (it is  $f_{\theta_{02}}^{p_{02}}$  in our example) rotated so that its major axis is perpendicular to the new local front's line segment. The new rate of spread and direction (same as major axis) of this new front are determined from the updated model parameters.

**Model forward propagation:** After updating the prior model, the Master  $S_0$  sends the updated model information to its helper-slave that detected the phenomenon most recently ( $S_2$  in our example), and asks it to become the new Master. If  $S_2$  satisfies the aforementioned necessary conditions for becoming a Master it accepts the request and returns a confirmation message to  $S_0$ . Upon receiving this confirmation,  $S_0$  sends a "release slaves" message to its neighbors and if they are not "enslaved" by some other Master(s) they change in turn their SSF to 0 (so that they may become Masters themselves in the future). If  $S_2$  does not satisfy the necessary conditions to become the new Master, it rejects the current Master's request and  $S_0$  tries the same negotiation with  $S_1$ . If  $S_1$  also fails to become a new Master,  $S_0$  asks both  $S_1$  and  $S_2$  to propagate the updated model information to their neighbors, releases the rest of its slaves and the algorithm repeats from the beginning.

### 3. MINIMIZING THE KL DIVERGENCE

We decided to model the evolution of a diffusing phenomenon by a modified 2D Gaussian function for the following reasons:

- Its elliptical shape, approximates reasonably well the evolution behavior of several phenomena of interest [5-8].
- Its parameters can be updated using closed form expressions. This is important since with our method we do not have to implement in sensor networks time and energy costly recursive optimization algorithms.

- The model information communicated from the Master to slave sensors can be summarized in a very concise form ( $\theta = \{A, \mu, \Sigma\}$ ). This also respects the stringent energy constraints of WSNs.

In essence model updating corresponds to finding the modified 2D Gaussian model which has minimum KL-divergence from the prior model and explains well the most recent sensor field measurements. Minimizing the KL-divergence to the prior model is justified since it is expected that in short time periods the local model parameters change smoothly.

The KL-divergence, also known as relative entropy, between two probability density functions  $f(x)$  and  $g(x)$ ,

$$D(f||g) = \int f(x) \log \frac{f(x)}{g(x)} dx \quad (2)$$

is commonly used in statistics as a dissimilarity measure between two densities [9]. For two d-dimensional normal distributions  $f$  and  $g$  the KL-divergence has a closed formed expression,

$$D(f||g) = \frac{1}{2} \left[ \log \frac{|\Sigma_g|}{|\Sigma_f|} + Tr[\Sigma_g^{-1}\Sigma_f] + (\mu_f - \mu_g)^T \Sigma_g^{-1} (\mu_f - \mu_g) - d \right] \quad (3)$$

We have shown (the proof is not included here due to space limitations) that the KL-divergence between two 2D modified Gaussians curves  $f, g$  as defined in (1) has a similar closed form,

$$D(f||g) = \frac{1}{2} \left[ 2 \log \frac{A_g}{A_f} - \log \frac{|\Sigma_g|}{|\Sigma_f|} - Tr[\Sigma_g^{-1}\Sigma_f] - (\mu_f - \mu_g)^T \Sigma_g^{-1} (\mu_f - \mu_g) + 2 \right] \quad (4)$$

and its absolute value satisfies the divergence properties [9]. In our application the functions  $f$  and  $g$  have the same origin point (have the same mean that is the location of a projection point on the local front (see section 2)) and thus the above formula can be simplified to:

$$D(f||g) = \frac{1}{2} \left[ 2 \log \frac{A_g}{A_f} - \log \frac{|\Sigma_g|}{|\Sigma_f|} - Tr[\Sigma_g^{-1}\Sigma_f] + 2 \right] \quad (5)$$

We have formulated and solved analytically (solution not shown due to space limitations) the following optimization problem to be stated here using the notation introduced in section 2.

**Problem:** Given a prior model  $f^{S_0}$  centered at projection point  $p_{0i}$ , find the parameters of the 2D modified Gaussian centered at the same point and having minimum KL-divergence from the prior model, i.e. Find:

$$\underset{\Lambda_i}{\operatorname{argmin}} \{ \operatorname{abs}(D(f_{\theta_0}^{S_0} || f_{\theta_{0i}}^{p_{0i}})) \}$$

Subject to the constraints:

*Equality constraints*

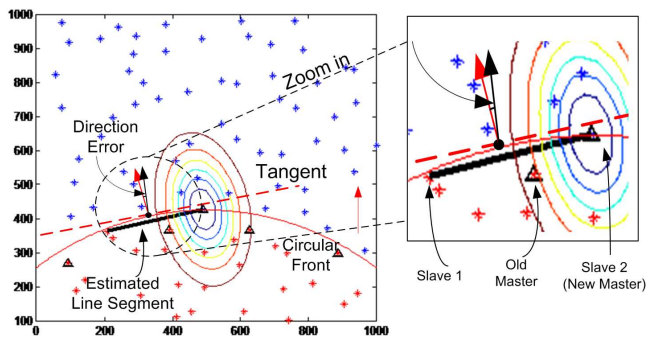
- $t_{0i} = \frac{A_0}{2\pi|\Sigma_i|^{\frac{1}{2}}} - \frac{A_0}{2\pi|\Sigma_i|^{\frac{1}{2}}} \exp(-\frac{d_i^2}{2\sigma_{11i}^2})$
- $\frac{A_0}{2\pi|\Sigma_0|^{\frac{1}{2}}} = \frac{A_0}{2\pi|\Sigma_i|^{\frac{1}{2}}}$

*Inequality constraints*  $\lambda_{11i} > 0, \lambda_{22i} > 0, |\Lambda_i| > 0$ , as required for a possitive definite  $\Lambda_i$  precision matrix where:

$$\Lambda_i = \Sigma_i^{-1} = \begin{bmatrix} \lambda_{11i} & \lambda_{12i} \\ \lambda_{21i} & \lambda_{22i} \end{bmatrix} \text{ and } \Sigma_i = \begin{bmatrix} \sigma_{11i}^2 & \sigma_{12i}^2 \\ \sigma_{21i}^2 & \sigma_{22i}^2 \end{bmatrix}$$

The first equality constraint is introduced to guaranty that the point which gives the specific value  $t_{0i}$  at a corresponding distance from  $p_{0i}$ , will fall on the major axis. The second equality constraint is a necessary condition for the absolute value of (4) to satisfy the divergence properties. Furthermore this constraint maintains the same (as the prior) prediction time span for the updated model.

The problem can be solved analytically to obtain algebraic expressions for updating the model parameters. (Due to lack of space the solution is not presented here).



**Fig. 3.** Matlab simulator snapshot. Red (blue) crosses denote sensors which have (have not) detected the evolving circular front. The line segment is updated by the the three sensors indicated by the arrows in the right side figure. The ellipsoid is the contour plot of the updated 2D Gaussian Model which describes the evolution behavior of the specific line segment.

#### 4. EXPERIMENTAL VALIDATION

In this section we present simulation results that demonstrate the ability of the proposed distributed WSN algorithm to estimate with reasonable accuracy the direction and speed of a propagating hazard and its robustness to sensor node failures. We conducted four simulations which differ only in sensor nodes failure probability. As failure we consider the inability of a sensor node to participate in the algorithm. The diffusing phenomenon front line was modeled as a growing circle, with a radius increasing at a predetermined constant rate ( $= 1.5$  meters/minute). The initial prior model speed parameter for all sensors was on purpose selected to be the one half of the "real" speed. Each simulation consisted of 50 runs. For each run 100 sensors were deployed pseudo-randomly within an area of  $1\text{km}^2$  and their communication radius was assumed to be  $R=150\text{m}$ . To study the behavior of the proposed WSN algorithm in the area of an evolving front we have developed a Matlab based WSN simulator. In order to evaluate the accuracy of the algorithm, we compared the local front estimates (in terms of direction and speed) to the known ground truth. To estimate the direction error we calculated the angle of two vectors: the vector that is perpendicular to the estimated front line segment and the vector perpendicular to the tangent of the circle (modeling the hazard) at the middle point of the corresponding arc (see Figure 3). For a model  $f_\theta$  ( $\theta = \{A, \mu, \Sigma\}$ ) the speed is estimated as the ratio  $\frac{\Delta s}{\Delta t}$  where  $\Delta s = 2\sigma'_{11i}$ , where  $\sigma'_{11i}$  is the variance element of the diagonal  $\Sigma'_i$  covariance matrix which results after rotating  $\Sigma_i$  such as the major axis of the corresponding 2D modified Gaussian is aligned with the horizontal x-axis. If we call the rotated model  $f'_\theta$  then  $\Delta t = f'_\theta(\Delta s)$ .

Table 1 provides for each simulation, the total number of model updates, the average number of messages exchanged per model update and the Mean and Inter-Quartile Range of the direction and speed estimation errors. The three rows of the table summarize the same results but under different failure probabilities for the sensor nodes. We observe that as the number of faulty sensors increases the number of model updates is reduced, as expected. However, the algorithm still maintains its ability to provide a smaller number of good quality local estimates of the evolving hazard parameters. This is evidenced by the insignificant change in error percentages in Table 1. We have also performed experiments where the true speed was time varying, or multiple hazards with different parameters are spreading in the same area and obtained similar error profiles (results not shown).

Probability of faulty sensor	Total # Updates	Mean Number of Messages per Model Update	Direction Error (degrees)		Speed Error %	
			Mean	IQR	Mean	IQR
0	1643	4.70	4.48	4.00	21.57	24.66
0.1	1397	6.47	5.51	5.68	18.18	23.33
0.2	1137	6.67	5.84	6.04	18.67	23.10
0.3	863	7.27	5.80	5.96	20.13	23.67

**Table 1.** Simulation results

#### 5. CONCLUSIONS

We presented a novel decentralized WSN algorithm which can track effectively the evolving local front of a hazardous diffusing phenomenon with a set of line segments. The spatiotemporal evolution of each segment is characterized by a modified 2D Gaussian function serving as a local predictive model. A WSN is used to update the parameters of the local models and propagate them in the direction of the front's motion in a fully decentralized manner. It can also estimate the time varying direction and speed of the phenomenon by forming dynamically clusters (triplets) of sensor nodes. The proposed scheme manages to accurately estimate the local front parameters even in cases where the initial prior model deviates significantly from the reality and/or a percentage of the deployed sensor nodes fail. This makes it suitable for dynamic data assimilation schemes used to calibrate periodically simulation models by exploiting field extracted knowledge as soon as it becomes available.

**Acknowledgements:** This research has been co-financed by the European Union (European Social Fund ESF) and Greek national funds through the Operational Program "Education and Lifelong Learning" of the National Strategic Reference Framework (NSRF) - Research Funding Program: Heraclitus II. Investing in knowledge society through the European Social Fund.

#### 6. REFERENCES

- [1] Ononye, A., Vodacek, A., and Saber, "Automated extraction of fire line parameters from multispectral infrared images". *Remote Sens. Environ.* 108:179-188, 2007.
- [2] J.Mandel, J. Beezley, J. Coen, M. Kim, "Data Assimilation for Wildland Fires: Ensemble Kalman filters in coupled atmosphere-surface models", *IEEE Control Systems Magazine* vol. 29, issue 3, 09, pp. 47-65.
- [3] J.Kim, K. Kim, S.Chauhdary, W. Yang, M. Park: "DEMOCO: Energy-Efficient Detection and Monitoring for Continuous Objects in WSN", *IEICE Trans. on Comm.*, vol. E91-B, pp.3648-3656.
- [4] W. Chang, H. Lin, Z. Cheng: "CODA: A Continuous Object Detection and Tracking Algorithm for Wireless Ad Hoc Sensor Networks". *Consumer Communications and Networking Conference* 2008, pp.168-174.
- [5] H. Anderson,"Predicting wind-driven wildland fire size and shape". USDA For. Serv. Res. Pap. INT-305, Feb. 1983.
- [6] M. Alexander, "Estimating the length-to-breadth ratio of elliptical forest fire patterns". *Proc. 8th Conf. on Fire and Forest Meteorology*, pp. 287-304.
- [7] M. Marghany,"RADARSAT for oil spill trajectory model", *Journal of Env. Modeling and Software*, Vol. 19, May 2004, pp. 473-483.
- [8] E.S. Manolakos, D. Manatakis, G. Xanthopoulos, "Temperature Field "Modeling and Simulation of Wireless Sensor Network behaviour during a spreading wildfire", *Proc. 16th EUSIPCO*, August 25-29, 2008.
- [9] S. Kullback, *Information Theory and Statistics*, Dover Publ. 1968.