

# Virtual Environment for Synthetic Smoke Clouds Generation

Angelo Genovese, Ruggero Donida Labati, *IEEE, Member*,  
Vincenzo Piuri, *IEEE, Fellow*, Fabio Scotti, *IEEE, Member*

Department of Information Technology  
Università degli Studi di Milano

Milano, 20122, Italy.

{angelo.genovese, ruggero.donida, vincenzo.piuri, fabio.scotti}@unimi.it

**Abstract**—In this paper we propose a virtual environment for the creation of synthetic wildfire smoke frame sequences, able to simulate a distant smoke plume and to integrate it with an existing frame sequence. This work provides a virtual tool to measure the accuracy of existing image-based wildfire smoke detection systems without the need to produce real smoke and fires in the environments. The proposed algorithm uses a cellular model driven by the rules of propagation and collision to simulate the basic physical principles of advection, diffusion, buoyancy, and the response to external forces (such as the wind). Adverse environmental conditions like fog and low-light are also simulated, together with the introduction of noise in order to reproduce acquisition defects. The resulting frame sequences are then evaluated by using a smoke detection system, which shows that our method for virtual smoke simulation gives results comparable to real situations. The extracted data can then be used to increase the performance of smoke detection systems when few real data are available.

**Index Terms**—Computer vision, smoke detection, lattice-boltzmann, simulation, virtual environment.

## I. INTRODUCTION

Fire and smoke detection systems based on sensor readings suffer from some disadvantages. First, they need to be sufficiently close in order to be effective, which can possibly result in a high number of nodes. Second, it is necessary to define a trade-off between power consumption and system reactivity: frequent measurements result in a more precise description, but they quickly drain the batteries. Moreover, a great number of sensors need a complex network architecture in order to communicate and relate information. Furthermore, some sensors, such as smoke sensors, give better results when they operate indoors and can be poorly effective in open environments [1].

The use of machine vision and image processing techniques for the purpose of fire and smoke detection has the advantages of a great distance vision, absence of latency and the possibility to extract more information (such as position, size, growth and kind of fire and smoke) [2]. Moreover, cameras can be used for other applications, such as intruder surveillance and control systems. The automation of fire and smoke detection requires ad-hoc techniques and algorithms, which need to be tested in many possible conditions. Ideally, the tests should be performed on real scenarios or, at least, a great number of different real frame sequences. However, it

is difficult to perform such tests in real situations and even to collect frame sequences regarding every possible situation. For these reasons, in this paper we propose an approach to virtually recreate frame sequences of different smoking scenarios, in order to test the effectiveness of smoke detection techniques in the greatest number of possible situations using synthetic data. The proposed approach is designed to simulate frame sequences related to vaste areas by implementing a simulation of the principles of fluid advection, fluid diffusion and buoyancy. The proposed algorithms are also constrained by the general physical laws of mass and momentum conservation. The model is completed with the simulation of pseudo-random effects in order to reproduce the apparently unconstrained behaviour of smoke dispersion, which can be caused by different factors (for instance, by the wind). The generated synthetic smoke plume is then merged with a real frame sequence and postprocessing techniques are used to simulate the effects produced by bad atmospheric conditions and low quality equipments.

The paper is structured as follows. Section II presents a literature review, regarding mathematical equations and models. Section III describes the proposed approach, and Section IV presents the experimental results.

## II. PREVIOUS WORK

### A. The equations of fluid flow

The physical model used in fluid dynamics is described by the Navier-Stokes equations, which treat the flow of a compressible viscous fluid in terms of a velocity vector field  $u$  [3]:

$$\nabla u = 0 ; \quad (1)$$

$$\frac{\partial u}{\partial t} = -(u \nabla) u - \frac{1}{\rho} \nabla p + \nu \nabla^2 u + f , \quad (2)$$

where  $u$  is the velocity vector field,  $\nu$  is the kinematic viscosity of the fluid,  $\rho$  is its density and  $f$  is an external force. The symbol  $\nabla$  is the vector of partial spatial derivatives. The Navier-Stokes equations are obtained by imposing that the fluid conserves both mass (Eq. 1) and momentum (Eq. 2). As described in [4], these equations can be solved considering the right-hand side contributions independently:

- 1) *External forces*: different forces, such as the buoyancy force or the force exerted by the wind, are represented by the term  $f$ ;
- 2) *Advection*: the advection term  $-(u\nabla)u$  describes how the velocity is transported;
- 3) *Diffusion*: the diffusion term  $(\nu\nabla^2u)$  considers how the smoke particles spread across the space;
- 4) *Pressure*: the pressure is described by the term  $\frac{1}{\rho}\nabla p$ .

In the case of a gaseous fluid (such as the smoke) the effects of the viscosity are negligible. Moreover, when the smoke velocity is well below the speed of sound, the compressibility effects are also negligible. These assumptions permit to simplify the equations as the incompressible Euler equations [5]:

$$\nabla u = 0 ; \quad (3)$$

$$\frac{\partial u}{\partial t} = -(u\nabla)u - \nabla p + f . \quad (4)$$

### B. Lagrangian and Eulerian Viewpoints

In order to describe the motion of a fluid (or a deformable solid) there are two possible approaches, named the Lagrangian and Eulerian viewpoints. The Lagrangian viewpoint models the system considering each particle with a position  $x(t)$  and a velocity  $u(t)$ . The Eulerian viewpoint, instead of tracking each particle, considers fixed points in the space and describe how the fluid quantities measured at those points (such as density, velocity, temperature) change in time. The function  $u(x, t)$ , which describes the value of the vector field  $u$  at the position  $x$  and time  $t$ , is based on the Eulerian viewpoint [6].

In this paper, we use the Eulerian viewpoint in order to consider the smoke as a spatial function  $d(x, t)$ , which describes the concentration of the gas at the position  $x$  and time  $t$ , within a fixed reference grid.

### C. Computational Fluid Dynamics and Computer Graphics

The two major areas which deal with fluid simulation are the Computational Fluid Dynamics (CFD) and Computer Graphics (CG). The applications of CFD are mainly concerned with the exact solution of the equations of fluid mechanics, in order to achieve a precise understanding of the phenomenon, especially for engineering and testing purposes. CG applications, however, try to simulate a realistic fluid motion and give importance to the real time simulation. In such applications, it is not mandatory to implement precise physical laws, but it is important to give the sense of reality and the ability to render such scenes as fast as possible (e.g. in computer games) [7].

In this paper, we focus more on the CG approach, since the exact simulation of physical laws would be unnecessarily complicated for the purpose of simulating distant smoke. CG applications of fluid simulation, however, often start with the implementation of numerical methods to solve the Navier-Stokes equations, such as the method proposed in [8]. This method is stable only for a sufficiently small simulation period of time  $\Delta t$ . An always stable approach for the solution of the equations is proposed in [4]. Some implementations are even available in the literature [9].

One of the problems of the numerical methods for the approximation of the Navier-Stokes equations is the numerical dissipation, which consists in an attenuation of fluid features, caused by the approximation errors introduced by finite-precision computers. For this reason, dissipation can cause a fluid more “passive” and less realistic. Two approaches for the solution of this problem are proposed in [10] and [7], with the method of “*Vorticity Confinement*”, which consists in feeding energy back into the systems in the form of vortexes.

### D. Lattice gas and lattice Boltzmann method

Other approaches exploit the lattice gas cellular automata (LGA) in order to simulate fluid flows. These approaches do not deal directly with the solution of the Navier-Stokes equations, instead they try to mimic the behaviour of the microscopic particles composing the fluid. The model consists in a lattice, in which each site can take a certain number of different states. Each state corresponds to a particle with a certain velocity. The evolution of the simulation is done in discrete time steps. After each instant of time, the state of a given site can be determined by the state of the site itself and the neighboring sites in previous instants of time. The state of each site is boolean and represents whether there is a particle that is moving upwards.

At each instant of time, two processes are carried out, propagation and collision. In the propagation process, each particle will move to a neighboring site determined by the velocity of the particle. In the collision process, rules are used to determine what happens if multiple particles reach the same site. These collision rules are required to maintain the total mass and to conserve the total momentum [11]. If the physical laws of mass and momentum conservation are respected, it is possible to derive the equations of Navier-Stokes [12, 13] by analyzing the interactions of such particles on a large scale.

Examples of clouds and smoke simulation using the lattice gas automata are proposed in [14, 15].

Lattice Boltzmann methods (LBM) are derived from the lattice gas model. The improvement consists in replacing the boolean values in the LGA model with real-valued densities of microscopic particles that move along each bond of the lattice, and in considering the motion as an average distribution of microscopic particles. From this model, it is also possible to derive the Navier-Stokes equations [16].

An implementation of the lattice Boltzmann method for gas simulation is described in [17].

## III. THE PROPOSED APPROACH

The proposed approach performs a visually realistic simulation of a distant smoke plume. Since at great distances most of the features of the smoke are not visible, our approach doesn't focus on a rigorous implementation of physical principles, but it is based on physical models and image processing techniques. The proposed approach can be divided in the following steps:

- 1) initial plume computation;
- 2) external forces computation;

- 3) velocity estimation;
- 4) merging with the frame sequence.

### A. Initial plume computation

The first step is based on the lattice Boltzmann method and consists in the initialization of a two-dimensional grid with a finite number of particles. The particles are then scattered throughout the grid with each frame.

The number of particles is a real number and is described by a space-time function  $f_i(x, y, t)$ , which describes the number of particles at the position  $(x, y)$  in the instant time  $t$ , with direction  $i$ . In our model, we consider only four possible directions ( $0^\circ, 90^\circ, 180^\circ, 270^\circ$ ). Each time step  $\Delta t$  consists in an image, and the evolution of the model generates a series of frames. The evolution is governed by the rules of propagation and collision [17]. The propagation rule can be written as follows:

$$f_i(x + e_{ix}, y + e_{iy}, t + 1) = f_i(x, y, t) , \quad (5)$$

where  $e_{ix}$  and  $e_{iy}$  are the  $x$  and  $y$  velocities components of the  $i$ -th direction, expressed in pixel / frame. The collision rule can be expressed as:

$$f'_i(x, y, t) = C_i \sum_i f_i(x, y, t) , \quad (6)$$

where  $f'_i(x, y, t)$  is the new number of particles with direction  $i$ , position  $(x, y)$ , and time  $t$ .  $C_i$  is the collision coefficient which represents the percentage of particles in the cell with direction  $i$  after the collision. The coefficient set  $C$  is modeled by a matrix, which is similar to the one presented in [15], and models the interaction with air particles by scattering the particles in the four possible directions, as:

$$C = \begin{bmatrix} 0 & 0.20 & 0 \\ 0.20 & 0.32 & 0.20 \\ 0 & 0.08 & 0 \end{bmatrix} . \quad (7)$$

The particles are more scattered towards higher regions than lower regions. It is possible to observe that smoke columns tend to be scattered by the resistance of the air and to rise relatively to their initial velocity. In order to guarantee the mass conservation, the following equation must be respected:

$$\sum_i C_i = 1 . \quad (8)$$

To simulate a plausible physical phenomenon by respecting the Navier-Stokes equations, momentum conservation must also be guaranteed. Combining Eq. 5 and Eq. 6, the global equation which describes the evolution of the model is:

$$f_i(x + e_{ix}, y + e_{iy}, t + 1) = C_i \sum_i f_i(x, y, t) . \quad (9)$$

Momentum conservation is verified by imposing a fixed velocity  $e_i$  for each particle in the lattice:

$$e_i = \sqrt{e_{ix}^2 + e_{iy}^2} = 1 . \quad (10)$$

In order to simulate the progressive disappearance of the particles, cells with a particle concentration lower than a predetermined threshold are not considered during the evolution of the model. This condition also helps to limit the computational complexity.

### B. External forces

The model of the smoke plume is completed by the introduction of external forces, such as wind and buoyancy. The wind is modeled by adding additional pseudo-random movements to the propagation equation (Eq. 5). In order to model the major wind force at greater heights, these movements are proportional to the distance of the particles from the smoke source. The movements  $m_x$  and  $m_y$  in the  $x$  and  $y$  directions are given by:

$$m_x = M d_x ; m_y = M d_y , \quad (11)$$

where  $d_x$  and  $d_y$  are the drift directions and  $M$  is the drift value, given by the equation:

$$M = \left( \frac{D}{H} \right)^2 \cdot f_d , \quad (12)$$

where  $D$  is the distance of the cell from the source of the smoke,  $H$  is the height of the smoke plume and  $f_d$  is a multiplying factor. Drift directions and probabilities are changed every  $N_c$  and  $N_p$  frames respectively.

The coefficient matrix  $C$  described in Eq. 7 caused the smoke plume to rise at each instant of time. After this initial rising step, we simulate a smoke cloud which moves mainly in horizontal directions by changing the scattering coefficients and drift directions. The new coefficient matrix which favours the scattering of the particles in the leftward direction can be expressed as:

$$C_L = \begin{bmatrix} 0 & 0.20 & 0 \\ 0.20 & 0.32 & 0.08 \\ 0 & 0.20 & 0 \end{bmatrix} . \quad (13)$$

Similarly, it is possible to derive the matrix  $C_R$  in order to favour the rightward direction.

The proposed approach also considers the effect of the buoyancy force on the smoke cloud. In the case of gaseous substances, such effect depends on the temperature of the gas, because the volume occupied by the gas is directly proportional to the temperature. It is possible to write the equation of the buoyancy force with respect to the temperature of the gas as described in [17]:

$$F_b = Hg(T_k - T_{ambient}) , \quad (14)$$

where  $F_b$  is the buoyancy force,  $g$  is the gravity in the vertical direction,  $H$  is the coefficient of thermal expansion, and  $T_{ambient}$  is the predefined ambient temperature value. The evolution of the temperature in the system is governed by the heat equation:

$$\frac{\partial u}{\partial t} - \alpha \nabla^2 u = 0 , \quad (15)$$

where  $u$  is the function  $u(x, y, z, t)$  that describes the variation of the temperature in a given region over the time [18]. We used a simplified model of the heat dispersion. Since temperature variations are similar to the gas particle scattering and it is plausible to simulate them by applying smoothing techniques to the smoke plume, our approach considers that the temperature is proportional to the particle concentration in each cell, and is subject to the same convection and diffusion. This approach is similar to the method proposed in [17], which uses a gaussian smoothing function in order to simulate the heat diffusion.

Assuming that the ambient temperature is constant, the buoyancy force (Eq. 14) can be modeled as a movement in the vertical direction, which is proportional to the number of particles in the cell:

$$B = \frac{\sum_i f_i(x, y, t)}{f_b}, \quad (16)$$

where  $B$  is the movement of the particles at the location  $(x, y)$ , and  $f_b$  is a proportionality factor.

### C. Velocity estimation

The simulation of slow moving smoke plumes is done by performing the operations of propagation, drift and the application of buoyancy force only at certain frame intervals. The contributions on the global velocity are expressible according to the formulas:

$$S_h = \frac{3}{N_c} + \frac{M}{N_d}; \quad (17)$$

$$S_v = \frac{3}{N_c} + \frac{M}{N_d} + \frac{B}{N_b}, \quad (18)$$

where  $S_h$  and  $S_v$  are the velocities in the horizontal and vertical directions, 3 represents the three applications of the propagation rule,  $M$  is the maximum drift value,  $B$  is the movement related to the buoyancy force, and  $N_c$ ,  $N_d$ ,  $N_b$  are three frame intervals which determine the time duration of the simulation phenomena. The greater this intervals are, the slower the resulting simulated smoke is.

### D. Merging with the frame sequence

The merging of the simulated smoke model with a real frame sequence is done frame-by-frame by adding the smoke intensity to the intensity of the pixel of the frame:

$$F(x, y, t) = R(x, y, t) + \frac{S(x, y, t)}{f_m} \quad \forall x, y, t, \quad (19)$$

where  $R$  and  $S$  are the real and simulated smoke sequences,  $f_m$  is a reducing factor used to control the visibility of the smoke and  $F$  is the resulting merged video. With positive values of  $f_m$ , the resulting smoke is a high albedo smoke, while negative values of  $f_m$  produce a low albedo smoke. This approach naturally solves the problem of blending the smoke with the background.

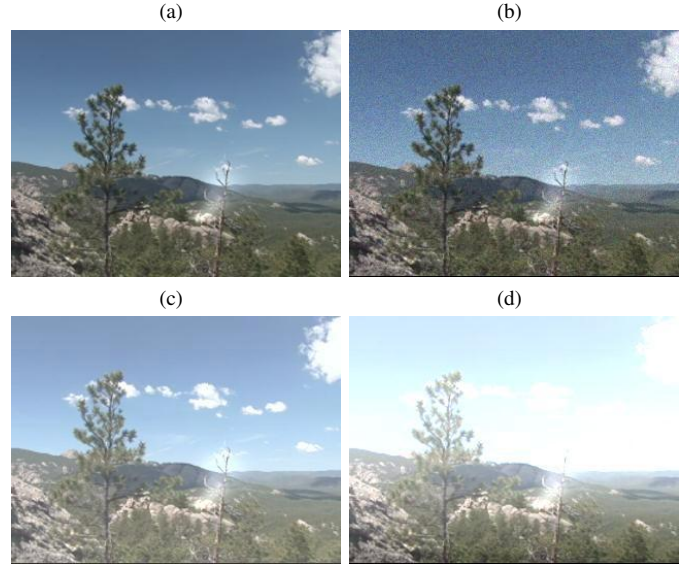


Fig. 1. Examples of simulated adverse conditions: (a) original image; (b) additive noise; (c) increased luminance; (d) fog.

### E. Simulating adverse conditions

Noise in the acquisition process is simulated by artificially adding Poisson-distributed noise generated directly from the image data, which visually gives the most realistic result (Fig. 1b).

Light effects simulating different seasons or moments of the day are added by increasing (or decreasing) the global luminance component of each image by a certain factor (Fig. 1c).

Fog effects are simulated assuming that lower regions of a frame are closer to the point of capture than higher regions, due to a perspective effect. Since fog is more visible in distant objects, we simulate the fog effect by superposing a gradient image over the real one, because the brighter regions of the gradient image are situated on the top part of the image (distant from the observation point). An example of simulated fog is shown in (Fig. 1d).

## IV. EXPERIMENTAL RESULTS

Smoke sequences are simulated by performing an initial plume computation, expansion, horizontal drift, and dispersion. The resulting sequences are then processed in order to simulate the adverse conditions described before. We used the same frame sequences to perform a comparison between the real and simulated data. We evaluated the obtained frame sequences by a visual inspection and by using a smoke detection software. Examples of virtual smoke simulation on real scenarios are depicted in Fig. 2.

The smoke detection software used for the evaluation of the obtained results is a pixel classifier based on neural networks, which uses features related to the movement, color and shape of the smoke plumes [19]. The system was tested with two different sets of frame sequences:

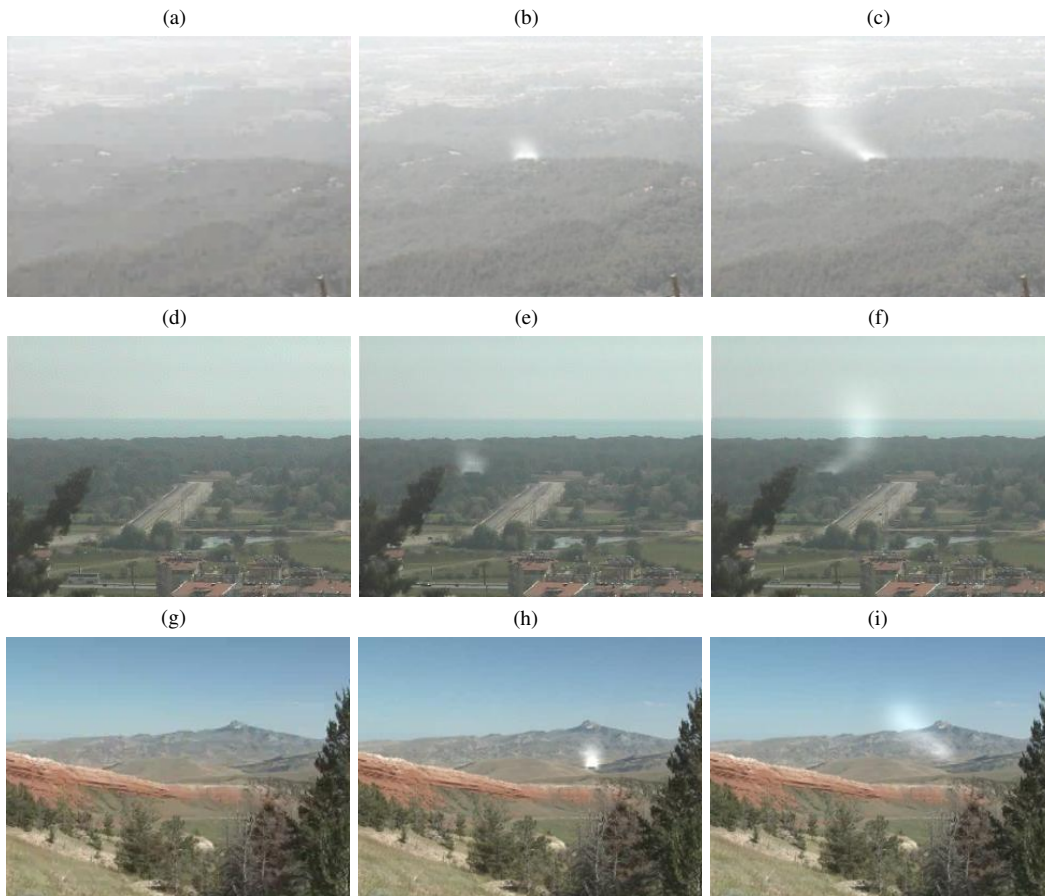


Fig. 2. Examples of synthetic smoke: (a),(d),(g) Scenes without smoke; (b),(e),(h) Scenes at time  $t = 0$ ; (c),(f),(i) Scenes after 200 frames.

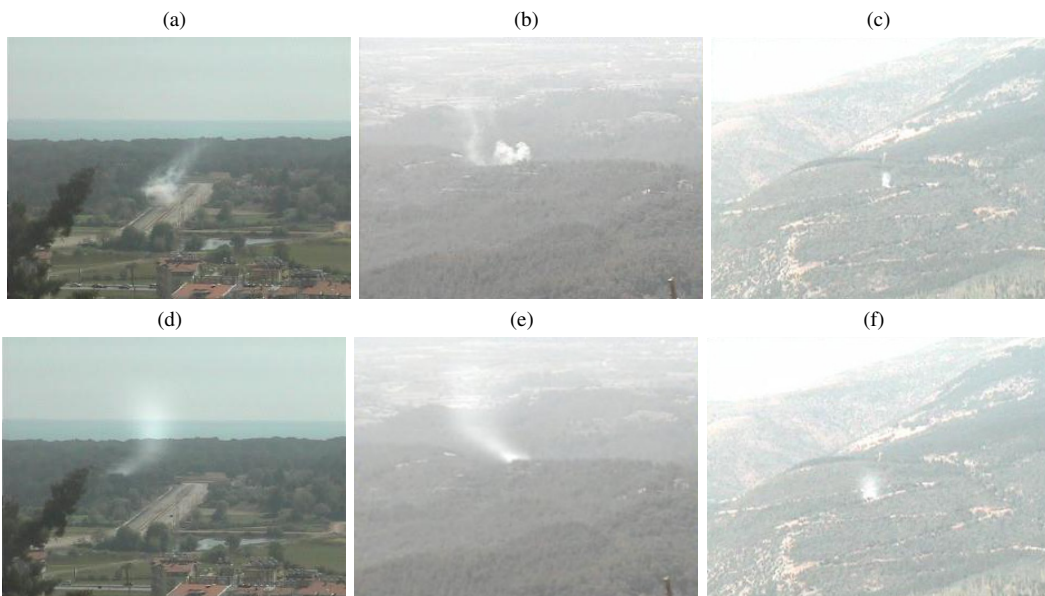


Fig. 3. Examples of real and synthetic smoke in the same conditions: (a,b,c) real smoke; (d,e,f) synthetic smoke.

- 1) *DSR*: collection of real smoke frame sequences;
- 2) *DSV*: collection of synthetic smoke frame sequences generated from non-smoke sequences of *DSR*.

Examples of real and synthetic smoke images are shown in

Fig. 3. We computed three feature datasets from these frame sequences: one related to the set of frame sequences *DSR*, one related to the frame sequences of *DSV*, and one related to all the frame sequences of *DSR* and *DSV*. The samples of

TABLE I  
SUMMARY OF DATASETS

Dataset Name	Frame Sequence	Features $\times$ Samples	Non smoke samples / Smoke samples
DSR	DSR	14 $\times$ 20318	19923/395
DSV	DSV	14 $\times$ 28207	27756/451
DSRV	DSR+DSV	14 $\times$ 48525	47679/846

TABLE II

RESULTS OF DETECTION ALGORITHM ON REAL AND SIMULATED SMOKE

Reference Method	Dataset Name	Tp (%)	Tn (%)	Fp (%)	Fn (%)	Sens. (%)	Spec. (%)	Total (%)
Neural	DSR	1.94	98.06	0.00	0.00	100.00	100.00	0.00
Approach	DSV	1.59	98.37	0.04	0.01	99.33	99.96	0.05
	DSRV	1.74	98.24	0.00	0.02	98.95	100.00	0.02

Notes: Sens. = Sensitivity; Spec. = Specificity.

the datasets correspond to the features related to each pixel of the frame sequences. The target values used in testing the classifiers were computed from the segmentation of the smoke region in every frame. The datasets are summarized in Table I.

Table II shows the results obtained by the smoke detection method using both real and simulated smoke frame sequences. We refer to the True Positives as the smoke samples correctly identified as such, and the True Negatives as the non-smoke samples correctly identified as non-smoke. It is possible to observe that the classification accuracy obtained by using the DSV dataset (composed by synthetic smoke samples) is comparable to the accuracy obtained using the DSR dataset (composed by real smoke samples). Moreover, the test performed by using the dataset DSRV (composed both by real and synthetic smoke sequences) shows that the frame sequences simulated by the proposed method can effectively be used in order to train smoke detection methods when few real data are available.

We also evaluated the proposed algorithms for the simulation of adverse environmental conditions by modifying the frame sequences of the dataset DSV and evaluating the performances of the method described in [19]. The results are summarized in Table III. It is possible to observe that the total classification error increases slightly, especially in the case of fog and low illumination. The proposed methods for the simulation of adverse environmental conditions, in fact, reduce the visibility of the smoke plumes.

The obtained results suggest that simulated data are useful to design fire and smoke detection systems. In fact, it is very difficult to capture a sufficient number of frame sequences, especially in adverse conditions.

## V. CONCLUSIONS

The paper presented a new method for the creation of realistic wildfire smoke plumes in order to easily assess the accuracy of smoke detection algorithms. The proposed method is based on the lattice Boltzmann model and it is governed by the rules of propagation and collision. The model is completed by simulating the effects of wind drifting and buoyancy. The simulation of virtual adverse conditions is also proposed.

TABLE III  
SUMMARY OF RESULTS UNDER ADVERSE CONDITIONS

Reference Method	Dataset Name	Tp (%)	Tn (%)	Fp (%)	Fn (%)	Sens. (%)	Spec. (%)	Total (%)
	DSV	1.59	98.37	0.04	0.01	99.33	99.96	0.05
Neural	DSV (+ Fog)	1.50	98.12	0.22	0.15	90.80	99.78	0.37
Approach	DSV (- Lum)	1.47	98.14	0.24	0.15	90.66	99.76	0.39
	DSV (+ Lum)	1.69	98.12	0.06	0.13	92.68	99.94	0.19
	DSV (+ Noise)	1.74	98.24	0.00	0.02	98.86	100.00	0.41

Notes: Sens. = Sensitivity; Spec. = Specificity; Fog = addition of simulated fog; -Lum: decreased luminance (Y) channel; +Lum = increased luminance (Y) channel; Noise = additive poisson noise.

We evaluated the realism of the obtained frame sequences by visual inspections and by using our smoke detection software, in particular performing a comparison between real and simulated smoke sequences under the same environmental conditions. The obtained results showed that the obtained frame sequences can be compared to real data. The data extracted from the synthetic frame sequences can then be used to increase the performance of smoke detection system when few real data are available.

## REFERENCES

- [1] S. Verstockt, P. Lambert, R. V. de Walle, B. Merci, and B. Sette, "State of the art in vision-based fire and smoke detection," *Proceedings of the 14th International Conference on Automatic Fire Detection*, vol. 2, pp. 285–292, 2009.
- [2] Z. Liu and A. Kim, "Review of recent developments in fire detection technologies," *Journal of Fire Protection Engineering*, vol. 13, no. 2, pp. 129–149, May 2003.
- [3] A. J. Chorin and J. E. Marsden, "A mathematical introduction to fluid mechanics," *Springer-Verlag. Texts in Applied Mathematics 4, Second Edition*, 1990.
- [4] J. Stam, "Stable fluids," *SIGGRAPH'99*, pp. 121–128, August 1999.
- [5] L. D. Landau and E. M. Lifshitz, "Fluid mechanics, 2<sup>nd</sup> edition," *Butterworth - Heinemann, Oxford*, 1998.
- [6] R. Bridson and M. Muller-Fischer, "Fluid simulation," *SIGGRAPH 2007 Course Notes*, August 2007.
- [7] D. Wanning, J. Limin, and J. Hongqiang, "A new method of smoke simulation," *ICENT 2010*, p. 267, June 2010.
- [8] N. Foster and D. Metaxas, "Modeling the motion of a hot, turbulent gas," *SIGGRAPH'97*, 1997.
- [9] F. Karlsson, "Smoke simulation," *TNM052 - Modeling & Animation, ITN/NVIS, Linkopings Universitet*, 2003.
- [10] R. Fedkiw, J. Stam, and H. W. Jensen, "Visual simulation of smoke," *SIGGRAPH 2001*, pp. 15–22, August 2001.
- [11] D. A. Wolf-Gladrow, "Lattice-gas cellular automata and lattice boltzmann models," *Springer*, 2000.
- [12] U. Frisch, B. Hasslacher, and Y. Pomeau, "Lattice-gas automata for the navier-stokes equations," *Physical Rev. Letters*, vol. 56, no. 14, pp. 1505–1508, April 1986.
- [13] S. Succi, "The lattice boltzmann equation for fluid dynamics and beyond," *Oxford Science Publications*, 2001.
- [14] Y. Dobashi, K. Kaneda, H. Yamashita, T. Okita, and T. Nishita, "A simple, efficient method for realistic animation of clouds," *SIGGRAPH'00*, pp. 19–28, 2000.
- [15] V. N. Vasyukov and A. N. Podovinnikov, "Simulating 2D images of smoke clouds for the purpose of fire detection algorithms adjustment," *IFOST 2008*, p. 369, 2008.
- [16] S. Chen and G. Doolean, "Lattice boltzmann method for fluid flows," *Ann. Rev. Fluid Mechanics*, vol. 30, pp. 329–364, 1998.
- [17] X. Wei, W. Li, K. Mueller, and A. E. Kaufman, "The lattice-boltzmann method for simulating gaseous phenomena," *IEEE Transactions on Visualization and Computer Graphics*, pp. 164–176, March-April 2004.
- [18] J. Cannon, "The one-dimensional heat equation," *Encyclopedia of mathematics and its applications, Addison-Wesley*, 1984.
- [19] A. Genovese, R. Donida Labati, V. Piuri, and F. Scotti, "Wildfire smoke detection using computational intelligence techniques," *IEEE International Conference on Computational Intelligence for Measurement Systems and Applications*, September 2011.