Walter W. Jones

National Institute of Standards and Technology, Gaithersburg, MD, US

Development of a multi-criteria algorithm for fast and reliable fire detection

Abstract

The purpose of detecting fires early is to provide an alarm when there is an environment which is deemed to be a threat to people or a building. High reliability detection is based on the supposition that it is possible to utilize a sufficient number of sensors to ascertain unequivocally that there is a growing threat either to people or to a building and provide an estimation of the seriousness of the threat. It has been shown to be possible to detect fires early and reliably using the analog signal of the current generation of fire detectors. The best combination for early detection has been shown to be the complement of ionization, photoelectric, carbon monoxide and temperature. This is "best" in the sense that it is possible, using current day sensors, to see characteristic signatures very early, as well as to deduce quantitative information beyond the normal tenability limits. This paper will demonstrate this with an example using a neural network trained with a model of fire growth and smoke spread.

Introduction

The purpose of detecting fires early is to provide an alarm consistent with an environment which is deemed to be a threat to people or a building. High reliability detection is based on the supposition that it is possible to utilize a sufficient number of sensors to ascertain unequivocally that there is a growing threat either to people or to a building and provide an estimation of the seriousness of the threat.

The current generation of fire detection systems[1] is designed to respond to smoke, heat, gaseous emission or electromagnetic radiation generated during smoldering and flaming combustion. Smoke is sensed either by light scattering or changes in conductive properties of the air, heat by thermistors, the electromagnetic spectrum by photodiodes and photovoltaic cells, and gas concentrations by chemical cells[2]. An important facet of the present work is utilization of sensors which are currently in use in fire detection systems, as well as those available from other systems, such as energy management and security. The information from the sensors themselves is analog data, measuring temperature, obscuration, species density, heat flux and other characteristics of the environment. What is needed is a means to provide earlier warning, and more useful information before and after alarm using these sensor suites.

Curve Matching Algorithms

Curve matching covers a wide range of mathematical techniques, from functional analysis to neural networks. Functional analysis is most useful when the signal to noise ratio is high[3] and one can match the signal to a specific curve of interest, for example, relating a t^2 signal to a heat release rate. Neural network analysis is useful when only the general shape of the curve is known and detail is not justified by the available signal. The regions 1, 2 and 3 in figure (1) show conceptually such a delineation. For all three regions, a pattern can be discerned. However, pattern matching is most usefully applied to the early, noisy signals in region 1 which does not lend themselves to definite statements of functional form, that is, when the signal-to-noise ratio is not high enough to provide a measure of the environment, typically S/N~2 to 4. Region 2 is the current range of available detection when point measurements provide sufficient signal to alarm, typically S/N~3 to 5. Region 3 is appropriate for signal extraction for fire following when the signal to noise ratio is typically greater than 10. We want to push detection capability into region 1, yet classify it correctly in terms of advice to the fire service or occupants.

Classification of fire types into low, medium and high likelihood consequences has implications for both fire service as first responders, and building maintenance personnel who might be able to fix problems before they rise to emergency status.

Figure (2) shows a typical sensor reading from a fire, carbon monoxide in this case. Detecting the presence of a fire traditionally has been to measure such signals, and provide an alarm when some condition is reached, for example, when the opacity is high or the carbon monoxide too high. Shown in the figure are alarm points for several detection strategies, an ionization detector, a photoelectric detector, and the CO*Ion algorithm discussed previously. The example is a surrogate for the range of signals which might be used for detection of fires[4]. Currently, temperature (T), opacity (OD), ionization (Ion) and carbon monoxide (CO) are the core signals we will focus on. In addition to these, carbon dioxide (CO_2), volatile organic hydrocarbons (VOC), nitrogen-oxygen compounds (NO), oxygen(O_2) and water concentration (RH) are possible future signals to incorporate.

An example of using pattern matching is discussed in the paper by Rose-Phersson et al.[5] The focus of the paper was the use a probabilistic neural network to combine signals from several transducers to reduce the likelihood of both false positives and false negative responses from detector systems. While this is similar to what we will use to reduce the time delay, the focus was on more reliable detection. The goal behind their work was to automate response to fires (e.g. sprinkler activation), so very high reliability is even more important than early detection. They demonstrated the optimal sensor set to be ionization, photoelectric, carbon monoxide and carbon dioxide, with temperature providing the best confirmation signal. In our case, we will work from the premise that the patterns we see will result in an alarm condition from the installed alarm base, so we want to respond as early as possible to these signals or patterns, in order to reduce the response time of the firefighters.

An Example of Implementation of a Neural Net Algorithm

An artificial neural networks (ANN) is a collection of mathematical models that emulate some of the observed properties of biological nervous systems and draw on the analogies of adaptive biological learning. The key element of the ANN paradigm is the structure of the information processing system. It is composed of a large number of highly interconnected processing elements that are analogous to neurons and are tied together with weighted connections that are analogous to synapses.

Learning in biological systems involves adjustments to the synaptic connections that exist between the neurons. This is true of ANNs as well. Learning typically occurs by example through training, or exposure to a “truthed” set of input/output data where the training algorithm iteratively adjusts the connection weights (synapses). These connection weights store the knowledge necessary to solve specific problems.

ANNs are good pattern recognition engines and robust classifiers, with the ability to generalize in making decisions about imprecise input data. They offer ideal solutions to a variety of classification problems such as speech, character and signal recognition where the physical processes are not understood or are highly complex. They are often good at solving problems that are too complex for conventional technologies (e.g., problems that do not have an algorithmic solution or for which an algorithmic solution is too complex to be found) and are often well suited to problems that people are good at solving, but for which traditional methods are not (you know a fire when you see it!).

The study of fire occupies a unique niche in the world of science and engineering because an unwanted fire is considered a failure in the sense that it is not a desirable outcome and is to be avoided. Detection and suppression are thus posed as means to avoid failure, which can be well characterized. For detection in particular, we have well defined failures which can be tested fairly reproducibly. In this highly regulated environment, in order for detectors to be approved for use they must detect fires as defined in UL 268 and EN 54 tests. In addition, there are nuisance criteria when the detectors should not alarm. While these latter are well recognized (dust, for example), there are no formal tests, though a simple negative (no fire) should in no case produce an alarm (a false positive). For the UL tests, there is a time prior to when the alarms should not activate.

The biggest difficulty in training neural networks is the extent of the training scenarios available. In fire research, the work has been limited to experimental data sets. Typically the training set consists of tens to hundreds of scenarios, while ANNs need tens of thousands to produce highly reliable classification. Using the fire model, CFAST, we can generate a very large set of training and testing scenarios.

For this example, we consider the use of single head (multisensor) detector in a single compartment. The sensor suite consisted of four sensors: oxygen, carbon monoxide, opacity and temperature. A more complete characterization would consider each sensor separately, as well as all combinations. This would provide a sense of the effect of losing a sensor (fault detection).

We have a model for fires which has been extensively tested, CFAST[6]. We used this model to generate training and testing scenarios which cover a very fine delineation of the event to be detected. Using such a model allows us to generate the tens of thousands to hundreds of thousands of examples necessary to provide sufficient training for a network.

The base case used: Standard atmosphere of 101,300 Pa, A single compartment of 13x13x2.4 m, Two cracks (one vertical, one horizontal) to account for leakage, One door of (0.9 x 2.3) m and One window of (0.9 x 1) m.

Starting with this base case, variations of the base case scenario were generated based on

(3) Ambient conditions: outside to inside temperature the same or ±15 ˚C
(3) Wind: none, into door (away from window if present) or away from door
(3) Fire size: (1, 10, 100) kw - note: no fire at all is a special case
(3) Position of fire: floor, and 0.5 m, 1.0 m above the floor in the center of the room
(4) Door width: open, ½, 1/4, 1/8 width
(2) Window: open or closed (0.9 m x 2.3 m)
(4) CO: (0.0, 0.001, 0.01 and 0.05) kg/kg or (0%, 0.1%, 1% and 5%) by fraction
(3) Smoke yield (optical depth): (0, 0.01, 0.05) kg/kg or (0%, 1% and 5%) by fraction
(2) Hydrogen carbon ratio in the fuel: (0, 0.2) kg/kg

This is 20,768 variations, which were then used to train the neural network. The scenarios were 300 second calculations with a time slice every 30 seconds. While this suite is sufficient to demonstrate the feasibility of training multisensor networks, a somewhat more comprehensive set of scenarios might include variable room size, non-rectilinear compartments and a range of radiative fraction, which would increase the number of scenarios, calculation and training time about an order of magnitude.

Three training exercises were performed: 1) a subset of the parameter space comprising 5000 scenarios, and 5000 for testing; 2) a complete set of scenarios (20726), and a small subset for testing (42) (total of 20768); and 3) preconditioning to supplement training for those cases when a fire is known to exist.

In order to be considered fast, the detection scheme must be at least fast as current detection algorithms. For high reliability, we are looking for means of seeing all real fire (no false negatives), and not responding to those deemed to be nuisances (no false positives). A metric for the former will be discussed as part of the analysis of results. The metric for false positives (nuisance alarms in the present context) and false negatives (missing a real fire), the scenarios are either fires or nuisance signals. Except for the base case of no heat release, which by definition is not a fire, the remainder are classified as real or nuisance by whether they pose a threat at any point in the curve to people or property. The classification is based on the ISO Toxicity Specification[7]. For exercises 2 and 3, of the total scenario space, 15 916 cases were fires and 4 852 non-fires. These latter (23%) are nuisance signals in the present context. A more complete classification scheme would further classify these according to Tables 1 through 3.

Mathematically, a neural network is a set of weight matrices which multiply sensor signals, and use a function (in our case a linear ramp) to combine the results. This provides a classification of data. Schematically, it is shown in figure (3), where **p** represents the measurement points, a vector of length R (in our case, this is the number of sensors), **b** a bias vector for the algorithm (always set to zero in our training), **w** the weight matrix (the answer so to speak). In the following training cases, we used R=4, but typically, it can range from 1 (a single sensor) to 9 (see ref.) which would be a very general multi-criterion sensor head.

The end point of such a system is a weight matrix which when multiplied by the sensor suite (**p**) produces a classification number; we used a simple classification of true or false (fire or non-fire). We trained a network with a single hidden layer of 10 neurons, and a single output layer using a linear transfer function. Thus we have only one matrix which needs to be adjusted. The training method used was Levenberg-Marquardt[8]. We have a set of four sensors, with 31 points (30 intervals). The data were presented to the learning algorithm, which modified the weight matrix (**w**) until a (defined) error level was reached.

We applied this technique using the Matlab[9] simulation tool, with the Neural Network Toolbox. Each data set was presented to network, and it adjusted the weight matrix. After completing the training, the network was presented test data, and classified the new sensor readings as a fire or non-fire event. Since we are concerned with a binary decision, the results were descritized to 0 or 1. In actuality, the data was a spectrum and additional training could be provided to further refine the classification scheme to non-fire, nuisance or significant event.

For the first case, there were no false positives or false negatives. That is, all fires were detected and no alarms when a fire did not exist. The time to alarm was generally the same for conventional detection and the trained network. The time to do the CFAST calculations was approximately 45 minutes, and the training time approximately 1 hour.

The time/temperature curve shown in figure (4) has the alarm points overlaid. The solid lines are example 1 and the dashed lines example 2. The vertical ticks are the corresponding detection time for conventional detection (green) and the neural net with training (red).

For the second example, all 20768 scenarios were used. In order to test the network, 42 of the 20768 scenarios were used for testing and not used for training. This then constituted a sampling of data which the network should be able to recognize. Of the forty two tested, there were no false positives (nuisance alarms), that is no fire detected when a fire did not exist; however, there was one false negative, not showing an alarm when a fire was present. This is about a 2% failure rate. The scenario which failed is marginal for the network, and to improve performance, the scenario suite needs to be extended to provide a finer resolution. In actual commercial detection systems, false negatives occur (3 to 20)% of the time[10] and false positives (30 to 50)% of the time, so we have improved on the detection capability as well as reduced the time to detection.

This training was done with a 10 neuron system. A systems with 20 neurons and two hidden layers was tried as well, without improvement. The time to detection for this second training example was always as early as conventional detection, as shown in figure (5). The time to do the CFAST calculations was approximately 2 hours, and the training time approximately 3 hours. The two cases shown, 007051 and 017658, are randomly picked from the 42 test cases.

For the third training example, the truth vector (when the fire exists) was preconditioned for those cases we know a fire will exist. For example, for the 100 kW source, it will at some time be considered a fire. For these cases we can set the training vector to “true” at after the first interval. Once again, there were no false positives and a single false negative (same case as before). The time labeled “preconditioned” in figure (5) was the response for the two cases shown in the figure for the example 2 testing regimen, 007051 and 017658, thus showing the value of using additional information in the training regimen.

This third training example takes advantage of the fire problem. We start with the scenarios. These produce curves of time, temperature, co, and so on. At some point we decide there is a fire. At the simplest level, used in 1 and 2, it is done the based on commercial detection schemes or the toxicity assessment discussed earlier. However, we can add to that information base, by noting that certain scenarios are going to be classified as fires, and tell the system from the beginning. For example, a 100 kW fire will must be detected, as must a 5 % CO condition. So for certain scenarios, one tell the system that it is a fire after the first interval. That gets factored into the weight matrix so that curves of similar shapes trigger an alarm very early. And even ones that are close do so. It is because we are matching curves (high precision) and not trying to get detailed information (high accuracy) that this technique is so appealing in this application.

Observations

There is additional work which needs to be done before this can be used in actual sensor suites: final testing for this case needs to include an example experiment such as the Smoke Detector Tests[11]. In addition, the standard qualification tests and a set of nuisance signals must be included. This latter will require an instrument transfer function, which can be measured using the FE/DE test apparatus[12]. Finally, the training suite ought to be extended to include the wide range of geometries which exist in practice rather than just those used for qualification testing. The training of a neural network should allow this extension and would improve the robustness of detection systems. This then allows one to include cases which currently cause alarms, such as steam, but are clearly not fires.

A further extension would be to go beyond the simple alarm/no-alarm classification we have done here and report on nuisance alarms as distinct from fires. Interestingly, a cursory inspection of the testing scenarios shows that the network is doing a reasonable characterization of the scenarios in terms of the type of fires. It is likely that this work could be extended to classification according to Tables 1 through 3. This is important in that a nuisance signal is often a precursor to more serious conditions. The prime example is the case of an oven (and even more commonly a toaster oven) which can develop the right conditions (and measurable effluent) but has a low level fire until a door is opened.

Conclusions

The full gamut of fire detection is possible utilizing currently available sensor technology. It has been shown that it is possible to detect fires early and reliably using the analog signal of the current generation of fire detectors. The best combination for early detection has been shown to be the complement of ionization, photoelectric, carbon monoxide and temperature. This is "best" in the sense that it is possible, using current day sensors, to see signatures very early, as well as to deduce quantitative information beyond the normal tenability limits.

The most useful of the algorithms studied is the curve matching concept embodied in neural network methods. In training such algorithms, it is important to use a sufficiently large set of training and testing samples so that that the algorithm is robust. We would expect a single experiment to provide very early detection for that single response curve. However, as the number of training sets is increased, incorporating variations in geometry and insult, the time to reliable detection increases. As the number of sensors used increases, we expect the detection time to decrease. The trade-off is in the necessity for using large (more than 10,000) sample sets. With a judicious use of modeling and experimental testing, this should not be a burdensome exercise. We have demonstrated the training of a neural network to shown that it is possible, including very early detection. Although we find a 2 % error rate with the present training regimen, this is still considerably better than current detection (3 to 30)% as well as methods proposed to date (2 to 10)%.

Table 1. Nuisance signals (low likelihood)

Hairspray, Nail polish remover , bleach, furniture cleaning agents, disinfectants

Toaster effluents - except as can be classified as incipient fires

Ovens, Boiling water, coffee, showers and other steam sources

Dust and sawdust, concrete dust, overcooked popcorn and other microwave products

Propane and kerosine heaters and stoves, candles, cigarettes and matches

Heating systems (furnace)

Table 2. Incipient (long time to disaster) fires

Toaster oven effluents, Welding and arc welding, Cook-top effluents, frying bacon

Smoldering mattress, chair or other cushion furniture: cotton, down

Table 3. Fires (prompt)

Open cellulose fires (crumpled newspaper)

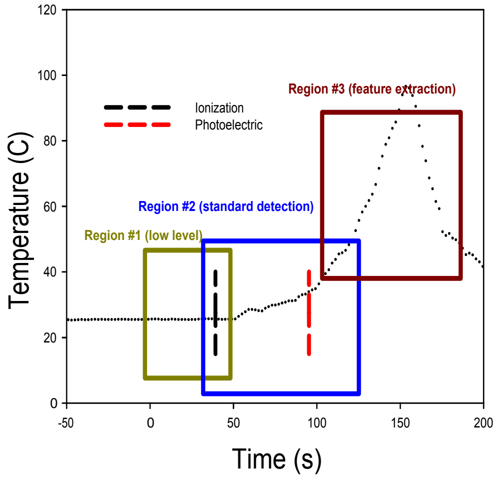
Flaming mattress, chair or other cushion furniture: cotton and foam

Liquid pool fire (heptane, gasoline, alcohol, paint thinner, acetone, vegetable oil)

Wood (wood based) furniture such as bookcases

Smoldering mattress, chair or other cushion furniture: foam

Power and signaling cables, Interior wall coverings such as wallpaper



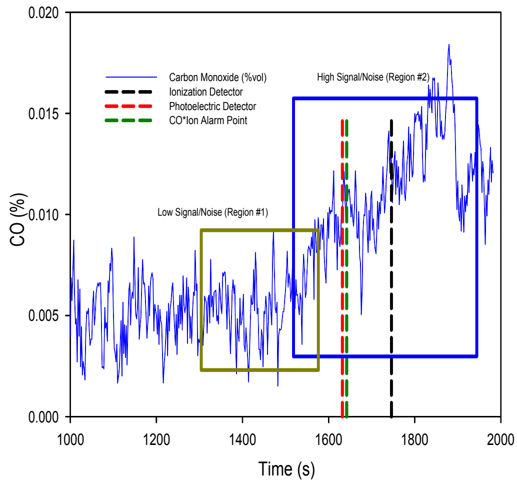
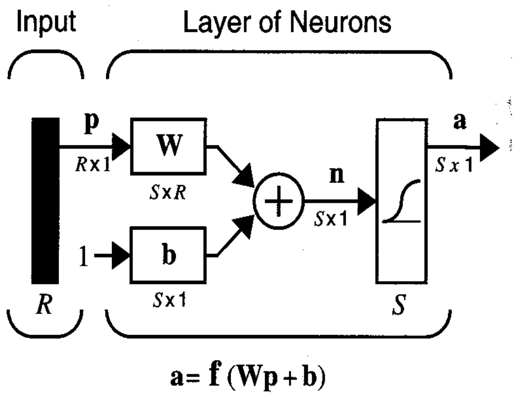


Figure 1. Delineation of detection regions for a flaming fire.

Figure 2. Carbon monoxide signal in SD 37, showing regions for curve matching algorithms.



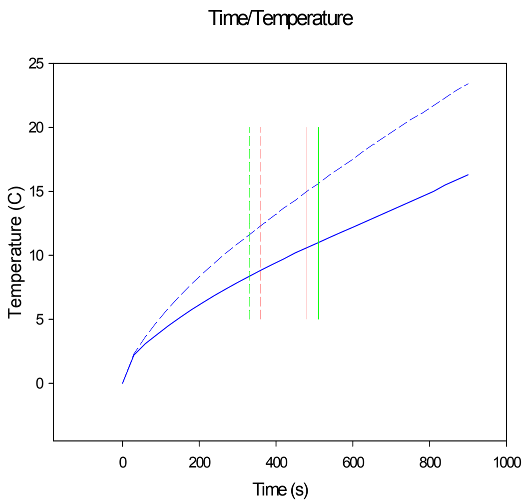


Figure 3. Schematic of a network layer.

Figure 4. Time/temperature curve for two of the 5000 test cases in example one.

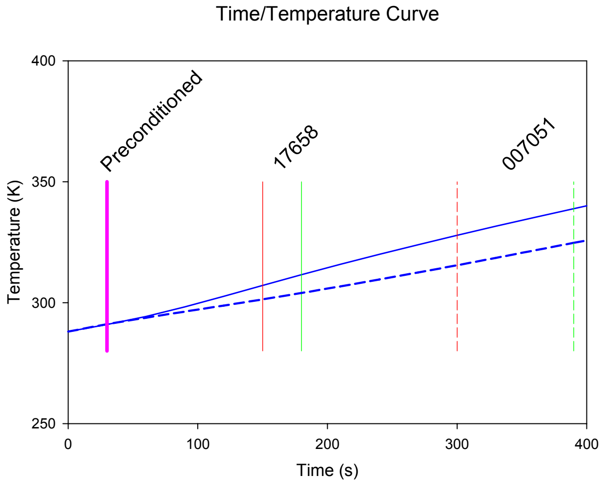


Figure 5. Time/temperature curves for two of the test cases for the second example. Green markers are for standard detection strategies and red for detection with the network. The marker labeled “preconditioned” is the result of these same test cases when the training vector is preconditioned for known fires.

References

1. A Review of Measurements and Candidate Signatures for Early Fire Detection, W.L. Grosshandler, NIST Internal Report 5555 (1995).

2. The Use of Surface and Thin Film Science in the Development of Advanced Gas Sensors, S. Semancik and R. Cavicchi, Applied Surface Science *70/71*, 337 (1993).

3. *Functional Data Analysis*, J.O. Ramsay and B.W. Silverman, Springer-Verlag, New York (1997).

4. Identification of Fire Signatures for Shipboard Multi-criteria Fire Detection Systems, D.T. Gottuk, S.A. Hill, C.F. Schemel, B.D. Strehlen, R.E. Shaffer, S.L. Rose-Pehrsson, P.A. Tatem, and F.W. Williams, NRL Memorandum Report, NRL/MR/6180-99-8386 (1999).

5. Multi-Criteria Fire Detection Systems using a Probabilistic Neural Network, S.L. Rose-Pehrsson, R.E. Shaffer, F.W. Williams, D.T. Gottuk, S.A. Hill and B.D. Strehlen, Sensors and Actuators, *B 69*, 325 (2000).

6. Technical Reference for CFAST: An Engineering Tool for Estimating Fire and Smoke Transport, W. Jones, G. Forney, R. Peacock and P. Reneke, NIST TN 1431 (2000).

7. ISO/TS 13571:2003, Life Threatening Components of fire - Guidelines for the estimate of time available for escape using fire data, International Organization for Standardization 1, rue de Varembé, Case postale 56, CH-1211 Geneva 20, Switzerland.

8. D. Marquardt, An algorithm for least squares estimation of non-linear parameters, SIAM J., 11:431 – 441, 1963; K. Levenberg. A Method for the solution of certain nonlinear problems in least squares. Quart. Appl. Math, 2:164 – 168, 1944.

9. The Matlab Development Environment, The MathWorks, Inc., Apple Hill Drive, Natick, MA 01760.

10. Analysis of Fire Signature Data Toward the Development of an Advanced Fire Detector, Progress Report on NIST Grant 1D0076 by J. Milke (September, 2001).

11. Performance of Home Smoke Alarms, Analysis of the Response of Several Available Technologies in Residential Fire Settings, R. Bukowsi, J Averill, R. Peacock, T. Cleary, N. Bryner, W. Walton, P. Reneke and E. Kuligowski, NIST TN 1455 (2003).

12. Fire Emulator/Detector Evaluator: Design, Operation, and Performance, T. Cleary, M. Donnelly, and W. Grosshandler, NIST SP 965, Proceedings of the 12 International Conference on Automatic Fire Detection, AUBE '01, Gaithersburg (2001); Towards the Development of a Universal Fire Emulator-Detector Evaluator, W. Grosshandler, Fire Safety Journal *29*, 113 (1997).