A Self-Organizing Map-Based Solution to the Automatic Detection of Meteor Echoes in Radio Spectrograms

Victor Ştefan Roman

Dept. of Automatic Control and Systems Engineering, Politehnica University of Bucharest (e-mail: victor.roman@acse.pub.ro)

Abstract: This paper aims at introducing a novel approach to the automatic detection of meteor echoes in spectrograms of radio recordings. The proposed meteor detection solution uses an artificial neural network to analyze data extracted from spectrograms and identify the meteor echoes present within. The success of the neural network based solution greatly depends on the network's architecture and training process. Several tests were performed to find the optimal neural network, while its training process was done using a manually built data set to insure the presentation of a sufficiently large set of data examples to the neural network. The final trained network is evaluated using a new set of spectrogram data and two distinct feature areas identification methods. The results obtained by this neural network were found to provide a statistical significant count of the meteor echoes in the BRAMS spectrograms, with correct data classification rates of over 88%.

Keywords: Automatic meteor detection, Self-organizing map, Multi-layered perceptron, Radio spectrograms.

1. INTRODUCTION

An important number of small-sized extraterrestrial objects enter Earth's atmosphere every day. These space objects, called meteoroids, interact with the atmospheric molecules and most will disintegrate due to ablation, a process which leaves behind a trail of plasma, referred to as a meteor. Studies have shown that between 4.4 and 7.4 tons of space material enter the upper atmosphere daily (Mathews et al., 2001). Observing meteors is an important research topic because it provides data into the meteoroids that fly through the atmospheric layers, such as physical properties or velocities. Through such studies, scientists attempt to obtain more information on the meteoroid streams, which is relevant for the better understanding of asteroids, comets and the solar system's evolution.

Typically, meteor observation is done using radio or video recordings systems, which are deployed as singular observatories or as multi-station networks. Such systems generate large amounts of data each day, which are, most often, manually analyzed by the scientists using the radio/video systems. The full automation of meteor observations is yet to be achieved, even though such systems would speed up scientific research and allow for the collection of data that may not be recorded otherwise. Steps towards the automation of radio/video observation of meteors have been taken through the implementation of automatic meteor detection solutions. Examples of automatic detection approaches in radio observations have been proposed in (Wan et al., 2005) and in (Roelandts, 2014), while automatic solutions for video observations are described in (Molau, 1999; Gural, 2007; Jenniskens et al., 2011 and Weryk et al., 2013). Also of interest in the field of meteor studies is the simultaneous radio and video observation of meteors, with an example of an automatic meteor detection solution described in (Weryk and Brown, 2012).

Artificial neural networks (ANNs) are mathematical models that attempt to imitate certain structures, characteristics and functions found in the human nervous system. They are used in a wide range of research areas to solve various modelling or pattern recognition problems. An ANN is a massively parallel model made up of simple processing units, called artificial neurons, which are grouped in interconnected layers. Each layer of neurons in an ANN has a specific role, with all ANNs typically having an input layer, where the input data is presented to the network, and an output layer, where the neural network's response is given. One or more other layers of neurons may be implemented in between the input and output layer in order to provide the neural network with data processing capabilities. A particular feature shared by all ANNs is that in order to achieve their design objective, they have to be subjected to a knowledge accumulation phase, called training. Through this learning process, the ANNs will be able to extract salient features or patterns in the input data. This knowledge is stored by the ANNs in the interneuron connections, also known as synaptic weights (Haykin, 1999). ANNs are known for their ability to work with high-dimensional data, to learn complex patterns, as well as for their ability to generalize in noisy environments.

Previous studies aimed at providing automatic meteor detection solutions based on ANNs have tackled two different meteor detection approaches. One automatic detection approach was built to search for meteor echoes in the time signal representation of radio recordings. This method, first described in (Roman and Buiu, 2014), and then expanded in (Roman and Buiu, 2015b), samples the radio recordings using a 0.1s long sampling window and then uses two types of ANNs, a multi-layer perceptron (MLP) and a manually-identified self-organizing map (SOM), to analyze each data sample and decide which contains a meteor echo and which doesn't. Furthermore, a software tool named MESCAL, aimed at training and testing the two types of ANNs used in the study, is proposed in (Roman and Buiu, 2015b). A second automatic meteor detection solution based on ANNs was described in (Roman and Buiu, 2015c). This approach detects meteor samples in spectrogram representations of radio recordings. To do so, the proposed approach samples the input data as slices containing either 4 or 5 vertical lines and then, after calculating the average radio signal power in each data sample, uses a MLP to detect which samples contain meteor echoes.

In the present study, a novel ANN-based approach is used to search in spectrogram data from radio recordings and detect the meteor echoes within. The proposed solution extracts data from the spectrograms in a very simple manner and uses these samples to train an ANN using an unsupervised training algorithm. Through this process, the ANN will cluster the input data based on the similarity between the input samples. The neural networks trained in this manner are then turned into a feature classification tool by attributing each data cluster to one of the feature classes used in this study. After the training and cluster labeling processes are finished, the ANN is used to analyze a new set of data and its classification abilities are tested. An early attempt at using a manually-identified SOM to detect meteor echoes using the same spectrogram sampling approach as the one used here is described in (Roman and Buiu, 2015a). The results in the present study, which expand those presented in (Roman and Buiu, 2015a), consist of: a larger, better performing manually-identified SOM, a new, automatically-identified SOM-based detection solution, as well as an in depth look at how the two proposed ANN-based detection solutions fare at detecting the meteor samples found in the test dataset.

This paper's main contribution is an ANN-based approach to automatic meteor detection in spectrograms of radio recordings. This original approach implements a simple spectrogram sampling technique, one that avoids using signal or image processing techniques, and trains a neural network, through an unsupervised process, to analyze the spectrogram data and detect the meteor echoes present inside.

This paper is organized as follows. In section 2, the radio recordings used are presented and the spectrogram sampling process is described. In sections 3 and 4, the neural networks are introduced and detailed. The testing of the ANN-based approach and the results obtained are presented in Section 5. And finally, the study's conclusions are drawn in Section 6.

2. RADIO DATA EXTRACTION

The radio recordings used to develop and test the meteor detection approach described in this study were recorded at BRAMS (Belgian RAdio Meteor Stations), the Belgian meteor detection network (Calders and Lamy, 2012). BRAMS is a radio-only system which uses the forward scattering of radio waves to detect meteor trails in the atmosphere. It is composed of one transmitting beacon located in Dourbes, Belgium, and 26 receiver stations spread all over Belgium. The beacon emits a purely sinusoidal radio wave at a frequency of 49.97 MHz and all receiver stations record over a 2.5 kHz bandwidth centred on this frequency. Radio data recorded by BRAMS is stored as 5-minute-long WAV (sound) files, but the default way to visualize it is through spectrograms generated by the BRAMS Viewer (Lamy et al., 2013).

A spectrogram is a visual representation of a signal's spectrum of frequencies as they vary over time. Spectrograms are represented as two-dimensional graphics, in which time and frequency are represented along the orthogonal axes, and the signal's power is represented as the intensity at a particular time and frequency, usually using colour coding. A spectrogram is obtained by calculating the fast Fourier transform of the original time signal. A typical BRAMS spectrogram is presented in Fig. 1.

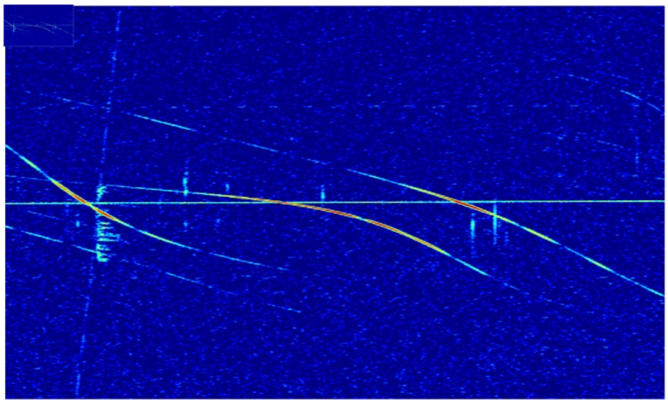


Fig. 1. Graphic representation of a typical BRAMS spectrogram.

The frequency range of the spectrogram is 200 Hz around the beacon's frequency. The central horizontal line is the beacon's directly received signal. The long-lasting, inverse S-shaped lines are airplane echoes. The short, vertical lines are radio echoes of underdense meteors, while the more complex shape to the left of the figure is an overdense meteor echo.

The reason for choosing to detect meteors in an automatically fashion using the proposed ANN-based approach in the BRAMS spectrograms is because spectrograms allow for an easy discrimination of the meteor echoes from other spurious echoes, such as the airplane reflections, as well as carrying information on the line-of-sight speeds of the meteors (Lamy et al., 2013). Due to the size of BRAMS spectrograms the data had to be sampled before it could be used with the proposed ANN-based detection approach. A rectangular sliding window was chosen to move across the BRAMS spectrograms and extract data samples. The size of the sampling window was chosen with respect to how most meteor echoes appear in BRAMS spectrograms, which is as short-lasting signals that have a broad frequency range. For this reason, a 30x20 pixels sampling window was used. The movement of this sampling window was designed to be done with an overlap of 10 pixels for a horizontal slide and an overlap of 15 pixels for a vertical slide. Lastly, the vertical average of each sample was calculated, thus obtaining a one-dimensional representation of the average power of the signal. One example of a spectrogram sample generated through this process can be seen in Fig. 2.

The meteor detection solution proposed in this paper uses an ANN to analyze and classify spectrogram data samples. Since the purpose of this study is to detect meteor echoes in spectrograms, it was decided that the spectrogram data be classified as either meteor or non-meteor. But in order for this approach to function, the ANN has to be trained first. To do this, a training set of data is required, one which contains sufficient examples of both classes of samples the ANN is expected to detect. Therefore, a number of 24 BRAMS spectrograms (2 hours worth of data) were sampled to build the training set. Additionally, an extra set of meteor samples was extracted from 96 other spectrograms (8 hours of data) and was included in the training data to provide even more examples of meteor data for the ANN to train with. In total, the training data set contained 767 meteor samples and 89359 non-meteor samples. The spectrogram data used in this study is available at (Roman and Buiu, 2015d).

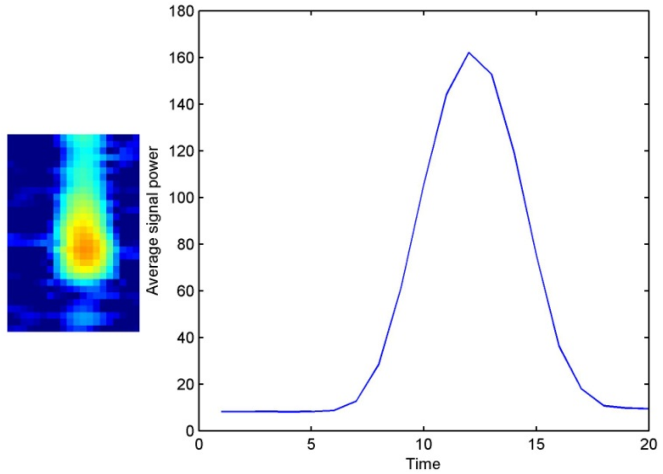


Fig. 2. Graphical representation of a meteor sample extracted for this study.

The figure on the left represents the 30x20 pixels sample, as it was extracted from the radio spectrogram. The figure on the right represents the final form of the data sample, obtained through the vertical averaging of the spectrogram sample.

3. SELF-ORGANIZING MAPS

The ANN used in this study to automatically detect meteors is the self-organizing map, which is also known as the Kohonen neural network (Kohonen, 2001). A SOM is a neural network based on competitive learning that generates a topographic map of the input patterns. SOMs are built as two-layered ANNs in which the output layer is formed as a one- or two-dimensional lattice of neurons, as seen in Fig. 3. Through the competitive learning process, the neurons in the SOM will selectively adapt to the various classes of patterns in the input data set, thereby creating a map over the output lattice in which similar input patterns are clustered in the same area of the map, while dissimilar input patterns will be clustered in different areas of the resulting map. The development of SOMs was motivated by the way the human brain maps different sensory inputs (e.g. acoustic, tactile, visual, etc.) onto different areas of the cerebral cortex in a topologically ordered way (Haykin, 1999).

The training of a SOM is the process which transforms the initial random distribution of neurons on the SOM's output lattice into a topographic map composed of areas where similar input patterns are clustered together. Thus, each area of the map can be considered a feature classifier. The training process, which takes place over many iterations (called epochs), uses a competitive learning algorithm during which the neurons in the output lattice compete to be activated by a particular input vector, with only one neuron in the lattice being activated at a given time. This competitive process is based on a discriminant function and the neuron which generates the largest value for this function will be the one to be activated. The adaptation of the neurons in the SOM is a process of cooperation, where a topological neighbourhood is determined around the neuron that won the competition for activation (called wining neuron) and all neighbouring neurons within this neighbourhood, as well as the winning neuron will have their synaptic weights modified to better resemble the input pattern.

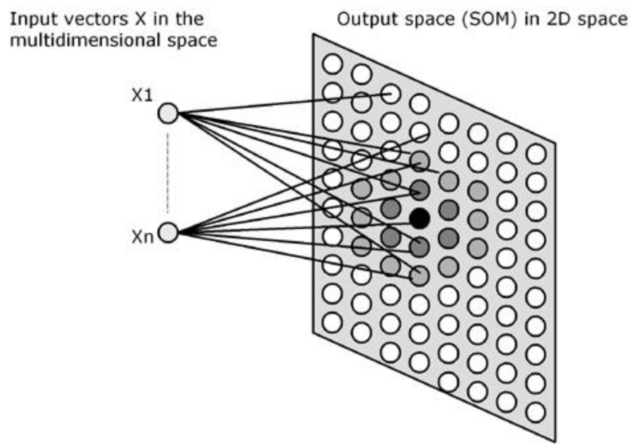


Fig. 3. A simple graphic representation of a self-organizing map.

The SOM training process can be summed up as follows:

1. Initialize the SOM and all neuron weight vectors.
2. Choose a random vector from the training set and present it to the neural network.
3. Calculate the discriminant function for all neurons in the lattice and determine the neuron which generates the largest value. This neuron will be called the best matching unit (BMU).
4. Determine the topological neighbourhood around the BMU and all neurons that are part of this neighbourhood.
5. Adjust the weight vectors of the BMU and all neighbouring neurons.
6. Repeat steps 2-5 for all the vectors in the training set.
7. Repeat steps 2-6 for a number of training epochs.

The initialization of a SOM refers to choosing initial values for the weight vectors $w_j(0)$ of the neurons in the output lattice. The only restriction here is that all weight vectors have to be different for all the neurons in the SOM.

For the competitive phase of the training process, an input vector is presented to all the neurons and a discriminant function is used to calculate the unit that best matches the input vector. For this study, the BMU is determined as the neuron whose weight vector is closest to the input sample in terms of the Euclidean distance, which is calculated as:

$$d(x, w_j) = \sqrt{\sum_i (x_i - w_{ji})^2}, \quad (1)$$

where x is the input sample and w_j is the weight vector of neuron j. Therefore, the competitive process aims at finding the neuron which has the smallest Euclidean distance to the input sample x. If we use i(x) to identify the best matching unit, then it can be determined as:

$$i(x) = \arg\min_j \left\| x - w_j \right\|. \quad (2)$$

Once the BMU has been determined, the SOM's training process will determine the neighbourhood $h_{j,i}$ around it and calculate which other neurons are part of this neighbourhood. A typical choice for the BMU neighbourhood, which is used in this study, is the Gaussian function:

$$h_{j,i}(t) = \exp\left(-\frac{d_{j,i}^2}{2\sigma^2(t)} \right), \quad (3)$$

where $d_{j,i}$ is the distance between the BMU i and the neuron j, while $\sigma(t)$ is the radius of the BMU's neighbourhood. The unique feature of the SOM training process is that it uses a shrinking neighbourhood, so that less neurons get trained as the epochs go by. The parameter that is modified every epoch to insure the shrinking of the BMU neighbourhood is the radius σ , as per:

$$\sigma(t) = \sigma_0 \exp\left(-\frac{t}{\tau_1} \right). \quad (4)$$

The parameter σ_0 is the value of the neighbourhood's radius at the start of the training process, τ_1 is a time constant, while t is the current epoch. Typically, the neighbourhood radius σ starts out quite large, sometimes as big as the entire network, but it ends up very small, to the point where only the BMU will be trained using the input vector.

The adaptation of the BMU and all neurons within its neighbourhood is done using the following rule:

$$w_j(t+1) = w_j(t) + \eta(t) h_{j,i}(t)(x - w_j(t)), \quad (5)$$

where $w_j(t)$ is the current weight vector of neuron j, $\eta(t)$ is the learning rate and x is the input vector. This equation shows that the weight vector's change through learning is directly proportional to the distance between a neuron and the BMU, therefore the closer a neuron is to the BMU, the more it will adapt to the input vector, while the BMU is the neuron to adapt the most. The training algorithm's learning rate $\eta(t)$, similarly to $\sigma(t)$, is a parameter that will shrink over time. This is done to control the level of change in a neuron's weight vector. While at first a larger $\eta(t)$ will lead to a larger change in a neuron's weights, towards the end, the ever smaller learning rate will only induce a small change to the weight vector. The learning rate $\eta(t)$ is modified through:

$$\eta(t) = \eta_0 \exp\left(-\frac{t}{\tau_2} \right), \quad (6)$$

where η_0 is the initial value of the learning rate, while τ_2 is another time constant.

After the training process is finished, the neurons in the output lattice will have formed a topographic map in which the vectors used for training have been clustered in several areas, with each such area corresponding to a class of features in the input data space. Through these areas of similarity, the SOM will be able to analyze new data and classify the input vectors based on which area they are mapped to. It should also be noted that the SOM training algorithm works in an unsupervised manner, where the user is not required to provide target classes for each input vector in the training set. Instead, the training algorithm will create feature classes by itself by clustering input vectors that contain similar patterns onto the same areas of the topographic map.

The process of training a SOM into a feature detector is reliant on a number of elements. The training data set has to contain a large number of examples of input patterns for all the feature classes defined for the given problem in order for the SOM to recognize similar patterns when new data will be processed by the network. Meanwhile, the number of neurons in the output lattice and the duration of training will influence the formation of areas of similarity on the resulting topographic map. Therefore, when designing a SOM, these three elements have to be properly chosen for the neural network to perform correctly. Aside from these factors, the quality of the SOM can also be assessed using two other metrics that will determine the network's ability to preserve the topological properties of the input data set. The first of these two metrics is the quantization error (Uriarte and Martin, 2008), which measures the average distance between each input vector and the best matching unit associated with it:

$$qe = \frac{1}{n} \sum \left\| x - bmu_x \right\|, \quad (7)$$

where n is the number of vectors in the input set, x is a data vector and bmu_x is its associated BMU. This error will determine how well the input data is mapped onto the SOM developed during training. A well trained SOM will typically generate the smallest quantization error, which means that the BMU neurons in the output lattice closely resemble their associated input vectors. The second metric analyzed is the topographic error. This error measure determines how well preserved is the topology of the input data by the SOM. The topographic error calculates the proportion of input vectors for which the BMU and the second BMU are not adjacent on the topographic map (Uriarte and Martin, 2008). The error is calculated as follows:

$$te = \frac{1}{n}\sum f(x), \tag{8}$$

where n is the number of input vectors, while f(x) is 1 if the first and second BMUs of vector x are adjacent, and is 0 otherwise. As with the quantization error, a small topographic error is indicative of a good SOM.

The optimal self-organizing map to be used for meteor detection was found after a series of tests done on SOMs trained with different values for the size of the output lattice and the duration of the training process. One goal of these tests was to reduce the quantization and topographic errors of the neural networks, as it was found out during this study that obtaining good values for these metrics is directly proportional to the ability of the SOMs to detect meteors. The other goal was to obtain a SOM that correctly classifies as many input samples as possible. The tests done during this study revealed that a SOM with 924 neurons in the output layer, displayed as a grid of 33x28 neurons, which was trained for 500 epochs, was the best performing neural network.

4. MULTI-LAYER PERCEPTRONS

The other type of ANN employed in this study, which was used to identify the feature areas on the topographic map, is a multi-layer feedforward network, also known as the multi-layer perceptron (Haykin, 1999). Typically, such a neural network is composed of an input layer of sensory neurons, one or more hidden layers of computational neurons and an output layer of neurons, where the network's response to an input signal is given. Being a feedforward ANN, all input data will be propagated through the network in a forward direction, from the input to the output layer. The signal that is generated at the output layer of a MLP is a result of very simple calculations done in the computational layers of the network: each neuron in such a layer calculates the weighted sum of all his inputs and passes this result to a nonlinear activation function, which determines that neuron's output signal.

Since the data processing and output generation in a MLP are directly influenced by the weight vectors of the neurons in the network, it becomes obvious that the performance of a MLP will benefit from having an appropriate set of weights assigned to each neuron in the network. The process through which a MLP is able to adjust the weights of its neurons in order to produce desired outputs is called the network's training. The MLP uses a supervised training process, where an error measure is propagated backward through the network, from the output to the input layer, in order to adjust the neurons' weights. The typical error measure used by an MLP is calculated as the difference between the network's actual and desired outputs:

$$e_j(t) = d_j(t) - y_j(t), \tag{9}$$

where $y_j(t)$ is the MLP's actual output and $d_j(t)$ is the desired output. The best known MLP training algorithm is the backpropagation algorithm (Rumelhart et al., 1986).

The training of a MLP is done over many epochs in order to gradually decrease the neural network's output error. During one training epoch, the MLP will be trained with all data vectors in the data set. Each vector is propagated through the neural network and it will generate a specific output signal. This signal is compared to that vector's desired output and the error thus generated will be propagated backward through the network to adjust the neuron weights in the MLP. The weight correction rule used by the backpropagation algorithm can be summed as (Haykin, 1999):

$$\Delta w_{ji}(t) = \eta\delta_j(t)y_i(t). \tag{10}$$

The term $\Delta w_{ji}(t)$ is the weight correction that will be applied to the link (synapse) between neurons j and i, η is the learning rate parameter, which tells how much the weights of the neurons will change, $\delta_j(t)$ is the local gradient and $y_i(t)$ is the output signal generated by neuron i, which is also the input signal received by neuron j.

In this study, a three-layer feedforward MLP is trained using the data extracted from the BRAMS spectrograms and with a set of desired outputs for each sample in the training set. Since the data analyzed in this study is to be classified as being either meteor or non-meteor, the MLP was designed with 20 neurons in the input layer, corresponding to the size of the spectrogram data vectors, and with 2 neurons in the output layer, one corresponding to meteor data and the other to non-meteor data. The number of neurons in the MLP's third layer (n_{hid}), the processing layer, and the number of epochs the training algorithm is run for (n_{epo}) are however independent of the problem at hand. Still, their values will greatly influence the network's performance, therefore they have to be chosen with great care.

Determining the optimal values of these two parameters was done using a procedure through which MLPs were trained with increasing values of one parameter, while the other one was fixed to a certain value. The scope of this process was to gradually decrease the mean square error value of the neural network to the point where for several values of the variable parameter, the MSE would not decrease any further. After one parameter was deduced through this procedure, its value was fixed to the optimal value and the other parameter's values were increased until the MSE of the MLP would not decrease for several different values of the variable parameter. Using this procedure, the optimal value found for n_{epo} was 184 epochs, while the optimal value for n_{hid} was deduced to be 211 neurons.

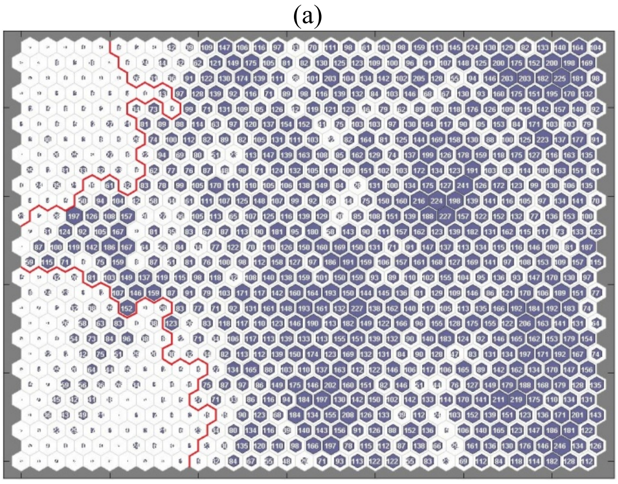
In the end, a multi-layer perceptron having the previously mentioned parameters was trained using the training data set extracted from spectrograms. This MLP was used to automatically determine the meteor and non-meteor areas on the topographic map built during the training of the optimal SOM.

5. METEOR DETECTION RESULTS

In order to test the SOM-based meteor detection solution proposed in this paper, a new data set was built using previously unused recordings from the BRAMS network. In total, 120 new spectrograms were sampled, which is the equivalent of 10 hours of recordings. Because evaluating the performances of the proposed detection approach requires that the samples in the test data set are known, a process of manual identification and labelling of these samples was done before the data was presented to the SOM. Through this process, a number of 1252 samples were identified as being meteor samples, while the other 447870 were labelled non-meteor samples. It should be noted that the even though sample labelling was done by a human operator, the test data set may contain some mislabelled samples.

Before the proposed SOM-based approach could be tested at classifying the spectrogram samples, a process of identifying the meteor and non-meteor areas on the map had to be done. Through this process, all neurons in the SOM's output layer are analyzed to identify which neurons have become, through training, meteor classifying units and which have become non-meteor classifiers. In this manner, meteor and non-meteor areas will be delimited on the topographic map, areas which can then be used to classify the test data into the two data classes used in this study.

(a)



(b)

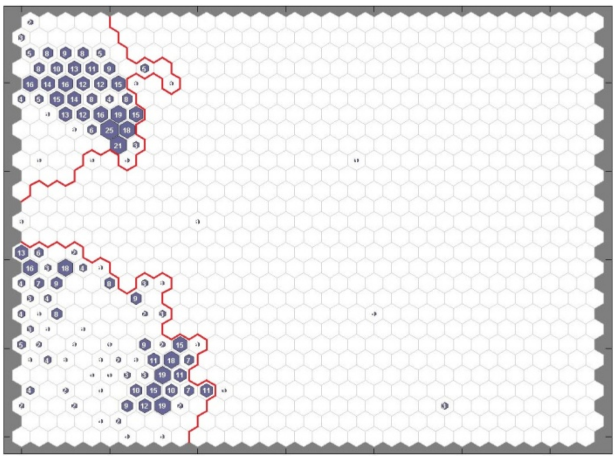


Fig. 4. Visual representations of two instances of the manuSOM.

The representation in 4(a) is the manuSOM's hits plot for the entire training data, while that in 4(b) is the hits plot for the meteor samples in the training data. Each hexagon in the two figures represents one neuron in the SOM's output lattice. The number in each hexagon represents the total number of input samples that were mapped to that particular neuron. In both figures, the red line is used to delineate the two meteor areas on the SOM (in the top and bottom left of the map). The rest of the SOM was labeled as a non-meteor area.

The first process of feature areas identification was done manually, after analyzing the topographic map that resulted through training. This was possible due to the fact that data in this study is divided in only two classes, meteor and non-meteor. It should be noted that for situations where the data is divided in more classes, a manual identification of the areas on the map would be much harder to do. In the case of this study, the area identification was done by analyzing not only the topographic map generated through training, but also the map obtained by presenting the SOM with only the meteor samples in the training set. These two maps, presented in Fig. 4, tell where the meteor data is mapped, as well as where the bulk of the non-meteor data is found. With this information, it was decided to separate the SOM into three areas, two meteor areas to the left of the map and a larger non-meteor area occupying the central and the right parts of the map, as can be seen in Fig. 4. Having obtained these two areas, the SOM could then be used to analyze and classify the samples in the test data set.

The results of the manually delimited SOM's (called manuSOM from now on) classification of the test data are found in Table 1. The results presented there, which were validated by hand, indicate that the proposed approach has a good ability to correctly classify spectrogram data samples, and is equally able at correctly classifying both types of spectrogram data. The overall rate of correctly detected samples obtained using the manuSOM is equal to 97.29%, which is a very encouraging result.

Table 1. The detection results obtained using the manually delimited SOM.

	Correctly detected		Incorrectly detected	
	#	%	#	%
Meteor samples	1219	97.36	33	2.64
Non-meteor samples	435745	97.29	12125	2.71

Because the approach proposed in this paper is intended to be a fully automatic solution to the detection of meteor echoes in spectrogram data, a new process of identifying the feature areas on the topographic map was done using the MLP trained with spectrogram samples. The SOM's output lattice contains neurons which, during the learning phase, adapt to the data in the training set by trying to become more alike to the patterns found in the input data. Therefore, the output neurons of a SOM can be analyzed with another classifying tool and be classified to one of the feature classes. For this study, the weight vectors of all neurons in the SOM's output lattice where presented to the MLP, whose classification results were used to associate each output neuron with one of the two data classes. In this manner, the feature areas on the topographic map were determined in an automatic fashion, as can be seen in Fig. 5, and the SOM could be used to classify the spectrogram data.

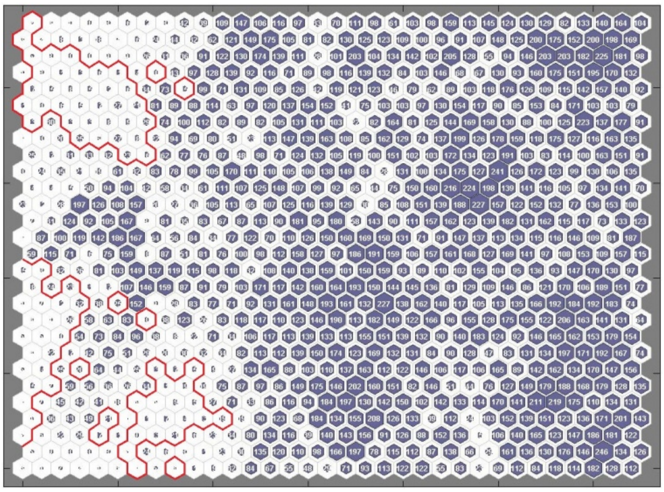


Fig. 5. Visual representation of the autoSOM's hits plot.

Delineated with red color are the feature areas that were identified by the MLP to be meteor areas. The rest of the SOM was classified as a non-meteor area.

The meteor detection results obtained using the automatically delimited SOM (called autoSOM from now on) are presented in Table 2. They show that the autoSOM also obtains a good detection score. The first observation that can be made is that this SOM is less able to detect meteor samples than the manually delimited SOM. At the same time, though, it can be seen that the second SOM is able to correctly classify more non-meteor samples and also has a higher overall correct classification rate of 99.63%.

Table 2. The detection results obtained using the automatically delimited SOM.

	Correctly detected		Incorrectly detected	
	#	%	#	%
Meteor samples	1102	88.02	150	11.98
Non-meteor samples	446357	99.66	1513	0.34

Analyzing the two SOMs tested in this paper, one can see that both perform really well at correctly detecting the samples extracted from the BRAMS spectrograms. The manuSOM is better at detecting the meteor samples in the test data, but the process of transforming it into a feature classifying tool is highly dependent on the operator's knowledge on the problem at hand and his ability to analyze the topographic map obtained through training the SOM. In comparison, the autoSOM, though less able to detect meteor samples, manages to correctly detect more non-meteor samples, and by doing so, has a better ratio of correctly detected test data samples. Still, it should be pointed out that the autoSOM's performances are highly dependent on using a well trained MLP. This is because the feature areas identification process for this case depends not only on the SOM having undergone a thorough training process, but also on the MLP's correct data classification ability.

The performance of the two SOMs tested in this study can also be compared using the confusion matrix of the detection results. This matrix is used to determine the percentage of correctly and incorrectly detected data samples from the total number of samples classified by the SOMs in each of the two classes. The results, included in Table 3, show that the autoSOM correctly detects a greater number of meteor samples from the total number of samples it has classified as being meteors. On the other hand, the manuSOM, although it correctly detects more of the meteor samples in the test data, has correctly classified only 9.14% of the total number of samples it has classified as meteor data. These results show that a solution which correctly detects more meteor samples is not necessary the better solution. In the case of this study, the manuSOM will generate a larger number of false counts compared to the autoSOM. As for the confusion results obtained by the two SOMs for the non-meteor data, it can be observed that the results are quite similar, with the autoSOM generating a larger number of misclassified data samples compared to the manuSOM.

Table 3. The confusion matrix for the two SOMs tested in this study. The "Total no. of samples" column contains the total number of samples that the SOM classified as being meteor or non-meteor.

		Total no. of samples	**Detection**	**#**	**%**
manuSOM	Meteors	13344	Correct	1219	9.14
			Incorrect	12125	90.86
	Non-meteors	435778	Correct	435745	99
			Incorrect	33	1
autoSOM	Meteors	2615	Correct	1102	42.14
			Incorrect	1513	57.86
	Non-meteors	446507	Correct	446357	99
			Incorrect	150	1

Analyzing the spectrogram samples included in the meteor data of the test set, one could see that the meteor samples are characterized not only by the shape of the signal, with a typical meteor shape being presented in Fig. 2, but also by the power of the received signal, as well as by the existence of non-meteor contributions, typically from airplanes, to the signal. Therefore, it is worth analyzing how the two proposed SOMs fare at correctly classifying the meteor samples in regards to their main characteristics. In order to do this study, the meteor data in the test set was divided into three classes.

One of the three classes, called the “Airplane” class, was built with all the spectrogram samples that contained both meteor and airplane radio echoes. This class is worth studying to analyze how airplane contributions in the data samples affect the classification abilities of the proposed SOMs. This is due to the fact that airplane echoes bring changes to the power of the radio signal in each sample, as well as modifying the shape of the received signal. An example of such a sample is presented in Fig. 6. Through manual analysis of the meteor data, 105 samples were extracted and included in this class.

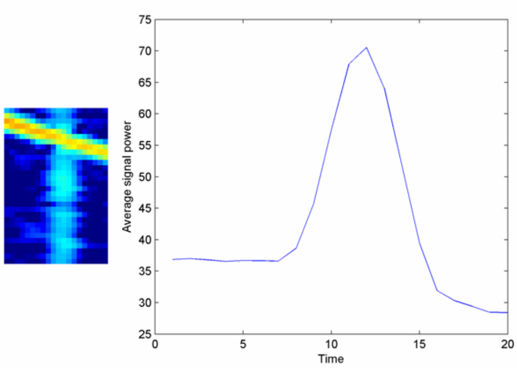


Fig. 6. Example of a meteor echo sample from the “Airplane” class.

It can be seen in the left figure how the airplane echo (the diagonal line) overlaps the meteor echo (the vertical line). The result of the airplane's contribution is shown in the final shape of the spectrogram sample in the right figure.

The other two classes studied were built based on the maximum power of the signal in each sample. To separate the data, a threshold was set at the 40 dB level. All samples whose maximum signal power was below this threshold were included in the class called “Faint”, while all samples with a maximum signal power equal to or greater than 40 dB were included in the class called “Strong”. It should be pointed out that the samples in these two classes contained no airplane echo contributions to the extracted radio signal. A graphic example of a sample from the “Faint” class can be found in Fig. 7, while an example of a sample from the “Strong” class is presented in Fig. 2. After analyzing the meteor data in the test set by hand, 473 samples were included in the “Faint” class, while 674 were included in the “Strong” class.

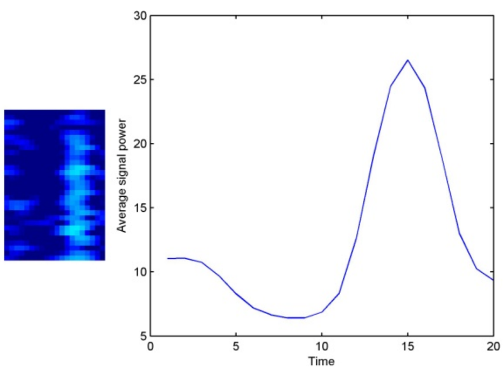


Fig. 7. An example of a meteor echo sample from the “Faint” class.

All three classes of meteor data were presented to both SOMs tested in this study, and the results are presented in Table 4. The manuSOM correctly classified more than 94% of the samples in the three classes, with the lowest result obtained for the meteors in the “Faint” class. The autoSOM, which has a lower meteor detection ability, as per Table 2, struggles visibly to correctly classify the samples in the “Airplane” class. The results for the other two classes are good, with the best results obtained for the samples in the “Strong” class.

Table 4. The detection results obtained by the two SOMs when analyzing the data samples in the three meteor classes.

		Total no. of samples	Correctly classified		Incorrectly classified	
			#	%	#	%
manuSOM	Airplane	105	100	95.24	5	4.76
	Faint	473	446	94.29	27	5.71
	Strong	674	673	99.85	1	0.15
autoSOM	Airplane	105	70	66.67	35	33.33
	Faint	473	380	80.34	93	19.66
	Strong	674	652	96.76	22	3.26

The first conclusion that can be drawn from this test is that both SOMs are able to detect meteor samples with a strong received radio signal. Apparently, the correct meteor detection ability of the ANNs is reduced by a decrease in the power of the received signal, with the autoSOM being more influenced by this factor. It is worth pointing out that the largest number of incorrectly classified meteor samples by each SOM is that of samples from the “Faint” class. Lastly, only the autoSOM appears to be underperforming in detecting samples from the “Airplane” class, while the manuSOM manages to correctly classify most samples in this class.

6. CONCLUSIONS AND FURTHER DEVELOPMENTS

An original approach to the automatic detection of meteor echoes in spectrograms of radio data recorded by the BRAMS network is presented and tested in this paper. The proposed solution implements a spectrogram sampling process that is operator independent and easy to reproduce. Data extracted from spectrograms is used to train a self-organizing map to recognize meteor samples, which is then manually processed to have its feature areas identified. A second neural network, a MLP, is trained with the same input data and then used to automatically identify the feature areas on the topographic map generated through the SOM's training.

The SOM trained in this paper, in both variants of feature areas identification, was tested using a new set of spectrogram samples. The results obtained during testing showed that the SOMs were able to correctly identify more than 88% of the test samples, while the overall detection rates were above 97%. But while both SOMs provided encouraging results, several differences were found between the two ANNs. Thus, while the manually delimited SOM was able to detect more of the meteor samples, the automatically delimited SOM correctly detected more overall samples from the test data and generated less false meteor sample detections. Furthermore, when analyzing the influence of the characteristics of the meteor samples over the SOMs' detection abilities, it was found out that while the manuSOM was able to detect the samples in the three meteor classes in an almost equal fashion, the autoSOM struggled to correctly detect the samples in the “Airplane” class, while also having a bit of trouble with the samples in the “Faint” class. It is worthwhile pointing out the samples in the “Faint” class of meteor data were the ones most misclassified by both SOMs, which hints that in the proposed form, the two SOMs are sensible to the power of the signal in the spectrogram samples.

In order to assess the performances of the meteor detection solution proposed in this paper, a comparison can be made with previously reported meteor detection results. When comparing the results presented in this paper to those of previous automatic detection solutions based on ANNs, such as those described in (Roman and Buiu, 2014; Roman and Buiu, 2015b, c), it can be seen that the detection approaches described in this study are able to provide equally good or better results. For example, the meteor detection solution described in (Roman and Buiu, 2015c), which also works with spectrogram data, is only able to correctly identify between 80 and 82% of the meteor samples and between 83 and 85% of the non-meteor samples, whereas the SOM-based solutions proposed here are able to detect 88% or even more, depending on the SOM used for detection, while non-meteor samples detection results are correct more than 97% of the time. The meteor detection solutions described in (Roman and Buiu, 2015b) use a manually-identified SOM, as well as a MLP used for detection, therefore they can be better compared to the results presented here. Comparing the two manually-identified SOMs, it can be seen that the SOM described here fares better at detecting meteor samples, 97% of the samples to the 91% reported in (Roman and Buiu, 2015b), as well as non-meteor data samples, with more than 97% to the 88.7% reported in (Roman and Buiu, 2015b). At the same time, it can be observed that while the automatically-identified SOM described in this paper manages to correctly identify less meteor samples than the SOM in (Roman and Buiu, 2015b), it fares much better at correctly detecting non-meteor data samples. Lastly, when comparing the results of the SOMs described here to the detection results obtained in (Roman and Buiu, 2015b) using a MLP, it can be noted that both ANNs in this paper have a better ability to correctly detect both meteor and non-meteor data samples than the MLP-based solution is able (86% of the meteor samples and 86.4% of the non-meteor samples).

Comparing the meteor detection results of the two SOMs presented in this paper with other, non-ANN-based detection approaches is, in ways, harder to do, as the methods of data sampling and meteor recognition vary greatly. Still, a brief look into other techniques can be taken in order to assess the abilities of the meteor detection solutions provided here. An automatic meteor detection solution working with radio recordings is the one described in (Roelandts, 2014). In a later study (Lamy et al., 2015) it is shown that this method is able to detect between 40% and 85% of the meteor echoes, depending on the parameters it uses. Still, in the same paper, it is shown that solutions that provide good meteor detection results also have a high percentage of misclassified non-meteor samples (i.e. which are classified as meteor samples). In this regard, the authors argue that for the given test, choosing a set of parameters which leads to a 65% meteor detection result will also lead to a small enough rate of false positive results. In a different paper, a meteor detection solution is proposed for working with video data. In that paper (Weryk et al., 2013), the authors report that their meteor detection solution is able to detect, based on the parameters it uses, between 67 and 90% of the recorded meteors. These results are also correlated to the number of well-tracked meteors, with the 67% meteor detecting configuration being able to better track more of the detected meteors than the 90% detecting configuration. And because the main interest of the authors was to have more well-tracked meteors, the decsion was to calibrate the system to the parameters of the 67% detecting configuration. If the results described in these two papers were to be compared to the results presented here, one could argue that the SOM-based solutions in this paper are better performing. These two results give an idea of what some requirements are in the field of meteor detection. It should be pointed out that both solutions described in this paragraph are able to detect whole meteor echoes and do not work with smaller data samples, therefore a direct comparison between those two approaches and the ones described in this paper cannot be made.

The results obtained by the SOMs proposed in this paper are closely related to the processes of identifying the feature areas on the topographic map. In this regard, the manual feature areas identification process would benefit from a more precise analysis and manual identification of the feature areas on the map, while the automatic identification process would be improved if a better performing MLP was used to determine the feature areas on the topographic map. The SOM's performances, as well as those of the MLP, are also influenced by the training data. Therefore, the two ANNs would benefit from an extended training data set, which would contain more examples of meteor and non-meteor samples. Also altering the performances of an ANN are the number of neurons in the network, or the duration of its training process. Thus, using different values for these parameters, or even adding extra layers of neurons, in the MLP's case, could maybe improve the meteor detection abilities of the neural networks.

In its present form, the ANN-based solution is used to analyze spectrogram samples obtained from one station in the BRAMS system and to classify these samples as being meteor or non-meteor data. As it is right now, the detection approach offers no other information on the meteor data it detects. One improvement that could be added would be to train the SOMs to discriminate which meteor samples were generated by underdense meteors and which were generated by overdense meteors. The proposed meteor detection solution may also be upgraded with a function that can extract the exact time moment in the original radio recording of each meteor sample detected using the SOMs.

Another development worth exploring would be to implement a multi-station system based on the meteor detection approach presented in this paper. This multi-station system would require analyzing how the SOM-based approach handles meteor echoes detected by multiple stations, as well as evaluating if such multi-station data would improve or damage the SOM's detection abilities. Developing a multi-station system may also require the training and implementing of distinct neural networks, one for each station in such a system.

ACKNOWLEDGEMENTS

This work was supported by a grant from the Romanian National Authority for Scientific Research, CNDI-UEFISCDI, project number 205/2012.

The work of V.Ş. Roman is supported by the Sectoral Operational Programme Human Resources Development (SOP-HRD), financed from the European Social Fund and the Romanian Government, under the contract number POSDRU/159/1.5/S/137390.

V.Ş. Roman would also like to acknowledge the help and kind support of Hervé Lamy, from the Belgian Institute of Space Aeronomy, BISA-IASB, and for providing the radio recordings used for this study.

REFERENCES

Calders, S. and Lamy, H. (2012). Brams: status of the network and preliminary results. *Proc. of the Int. Meteor Conf., IMC 2011,* 73-76, Sibiu, Romania.

Gural, P.S. (2007). Algorithms and software for meteor detection. *Earth. Moon. Planets.,* 102: 269-275.

Haykin, S. (1999). *Neural Networks: A Comprehensive Foundation, 2nd Edition*. Prentice Hall, Upper Saddle River, NJ, USA.

Jenniskens, P., Gural, P.S., Dynneson, L., Grigsby, B.J., Newman, K.E., Borden, M., Koop, M., and Holman, D. (2011). CAMS: Cameras for Allsky Meteor Surveillance to establish minor meteor showers. *Icarus,* 216(1): 40-61.

Kohonen, T. (2001). *Self-Organizing Maps, 3rd Edition*. Springer-Verlag, Secaucaus, NJ, USA.

Lamy, H., Gamby, E., Ranvier, S., Geunes, Y., Calders, S., and de Kaiser, J. (2013). The BRAMS Viewer: an on-line tool to access the BRAMS data. *Proc. of the Int. Meteor Conf., IMC 2012,* 48-50, La Palma, Spain.

Lamy, H., Anciaux, M., Ranvier, S., Calders, S., Gamby, E., Martinez Picar, A., Verbeeck, C. (2015). Recent advances in the BRAMS network. *Proc. of the Int. Meteor Conf., IMC 2015,* 171-175, Mistelbach, Austria.

Mathews, J.D., Janches, D., Meisel, D.D., and Zhou, Q.-H. (2001). The micrometeoroid mass flux into the upper atmosphere: Arecibo results and a comparison with prior estimates. *Geophys. Res. Lett.,* 28(10): 1929-1932.

Molau, S. (1999). The meteor detection software MetRec. *Proc. of the Int. Meteor Conf., IMC 1999,* 9-16, Stará Lesná, Slovakia.

Roelandts, T. (2014). Meteor Detection for BRAMS Using Only the Time Signal. *Proc. of the Int. Meteor Conf., IMC 2014,* 197-200, Giron, France.

Roman, V.Ş. and Buiu, C. (2014). Automatic detection of meteors using artificial neural networks. *Proc. of the Int. Meteor Conf., IMC 2014,* 122-125, Giron, France.

Roman, V.Ş. and Buiu, C. (2015a). A self-organizing map-based approach to automatic meteor detection in radio spectrograms. *Proc. of the Int. Conf. on Electronics, Computers and Artificial Intelligence (ECAI),* vol. 7, S-75-S-78, Bucharest, Romania.

Roman, V.Ş. and Buiu, C. (2015b). Automatic Analysis of Radio Meteor Events Using Neural Networks. *Earth Moon Planets*, 116(2): 101-113.

Roman, V.Ş. and Buiu, C. (2015c). Automatic detection of meteors in spectrograms using artificial neural networks. *Proc. of the Int. Symp. Applied Computational Intelligence and Informatics (SACI),* 131-134, Timişoara, Romania.

Roman, V.Ş. and Buiu, C. (2015d). BRAMS Radio Spectrograms and Spectrogram Samples for Automatic Detection of Meteor Echoes, Doi: 10.5281/zenodo.17394.

Rumelhart, D., Hinton, G.E., and Williams, R.J. (1986). Learning representations by back-propagating errors. *Nature,* 323: 533-536.

Uriarte, E.A. and Martin, F.D. (2008). Topology preservation in SOM. *World Academy of Science, Engineering and Technology,* 2(9): 867-870.

Wan, C.H., Doherty, J.D., Mathews, J.D. and Janches, D. (2005). Meteor detection and non-periodic bursty interference removal for Arecibo data. *J. Atmos. Sol.-Terr. Phys.,* 67(3): 275-281.

Weryk, R.J. and Brown, P.G. (2012). Simultaneous radar and video meteors - I: Metric comparisons. *Planet. Space. Sci.,* 62(1): 132-152.

Weryk, R.J., Campbell-Brown, M.D., Wiegert, P.A., Brown, P.G., Krzeminski, Z., and Musci, R. (2013). The Canadian Automated Meteor Observatory (CAMO): System overview. *Icarus,* 225(1): 614-622.