DETECTING GAS VAPOR LEAKS THROUGH UNCALIBRATED SENSOR BASED CPS

Diaa Badawi[*] *Sule Ozev*[†] *Jennifer Blain Christen*[†] *Chengmo Yang*[‡] *Alex Orailoglu*[◇] *A. Enis Çetin*[*]

[*] Department of Electrical & Computer Engineering, University of Illinois at Chicago, Chicago, IL
[†] School of Electrical, Computer and Energy Engineering, Arizona State University, Tempe, AZ
[‡] Department of Electrical & Computer Engineering, University of Delaware, Newark, DE
[◇] Department of Computer Science & Engineering, University of California, San Diego, La Jolla, CA

ABSTRACT

While Volatile Organic Compounds (VOC) and ammonia have a place in our daily lives, their leakage into the environment is harmful to human health. In order to prevent and detect gaseous leaks of harmful VOCs, a cyber-physical system (CPS) comprised of ordinary people or first responders is proposed. This CPS uses small, low-cost sensors coupled to smart phones or mobile devices with the necessary computation and communication capabilities. The efficacy of such a CPS hinges on its ability to address technical challenges stemming from the fact that identically produced sensors may produce different results under the same conditions due to sensor drift, noise, or resolution errors.

The proposed system makes use of time-varying signals produced by sensors to detect gas leaks. Sensors sample the gas vapor level in a continuous manner and time-varying sensor data is processed using deep neural networks. One of the neural networks (NN) is an energy efficient Additive Neural Network (AddNet) which can be implemented in host devices. The second NN is the discriminator of a GAN and the third a regular convolutional NN. AddNet produces comparable VOC gas leak detection results to regular convolutional networks while reducing area requirements by two thirds.

Index Terms— VOC gas leak detection; sensor drift; additive, convolutional, and generative adversarial (GAN) neural networks

1. INTRODUCTION

Volatile organic compounds (VOCs) and ammonia can be harmful to human health. If their indoor concentration exceeds a certain level, they may trigger asthma and rhinitis. VOC gases are a major contributor to global warming. Additionally, some VOC compounds such as benzene and toluene are carcinogenic. High concentrations of ammonia pose a health hazard as well as cause harm to the skin, eyes, and lungs [1–4]. Exposure to ammonia can occur from a deliberate terrorist attack or through an accidental release from farms or industrial and commercial facilities.

To prevent and detect gaseous leaks of harmful VOCs, we propose to develop a cyber-physical system (CPS) comprised of ordinary people or first responders using small, low-cost chemical sensors coupled to smart phones or mobile devices with the necessary computation and communication capabilities. Chemical sensors sample open air in a continuous manner and alert the first responders whenever they detect VOC and ammonia gas vapor leaks. Our goal is to detect VOC gas leaks and other dangerous high gas vapor concentration levels using un-calibrated low-cost sensors attached to smart mobile devices.

Chemical gas sensors constitute a potentially low-cost and practical alternative to conventional but expensive optical devices such as gas chromatographs and Medium Wave InfraRed (MWIR) cameras. However, lack of stability of measurements (which is called *sensor drift*) is an important problem in chemical sensing. Selectivity and sensitivity of chemical sensors decrease over time. As a result, identically manufactured chemical sensors may generate significantly different results when exposed to the same analyte under identical conditions [5–9].

Sensor drift is mainly attributed to two types of reasons, namely, the physical changes in the structure of the sensor and the operating environment. The so-called *first-order drift* stems from aging and "poisoning", an irreversible binding due to external contamination. The so-called *second-order drift* is due to external and uncontrollable parameters such as temperature and humidity variations in the environment. As a result, it may not be a good idea to set a constant threshold to detect the existence of a gas leak or other dangerous substances using low-cost chemical sensors. Such sensors are additionally impacted by changes in the concentration of the gas in open air due to wind.

Some VOC compounds and ammonia absorb infrared light at Medium Wave InfraRed (MWIR) and Long Wave InfraRed (LWIR) bands. LWIR sensors are also able to detect ammonia gas leaks in open air. However, the distance between the sensor and the source, and infrared reflections from the background significantly affect the recorded level [10,11]. Similarly, we can also visualize the existence of VOC gas vapor using Medium Wave InfraRed (MWIR) cameras as shown in Fig. 1. While in open air it is not possible to detect the concentration of the gas using MWIR and LWIR sensors, it is possible to record a time-varying signal and detect the existence of the gas leakage by naked eye or using a machine learning algorithm such as a neural network.

In this paper, we analyze the time-varying sensor signals using convolutional, additive neural networks and the discriminator network of a GAN to detect and classify VOC gas leaks and other dangerous gas emissions. The proposed analysis is applicable to both Chemically-sensitive Field Effect Transistors (ChemFETs) and Electrochemical Impedance Spectroscopy (EIS) sensors as they both produce time-varying signals.

The rest of the paper is organized as follows. Section 2 describes the machine learning algorithms used in this paper. Section 3 presents experimental results. We use an infrared data set and a publicly available chemical sensor drift data set obtained at the University of California at San Diego (UCSD) [8]. The paper finishes by offering a brief set of conclusions in Section 4.

This work is being supported in part by NSF grants 1739396 (UIC), 1739451 (ASU), 1739390 (UD) and 1739684 (UCSD) . The authors, Badawi and Cetin, additionally thank Nvidia for an equipment grant.

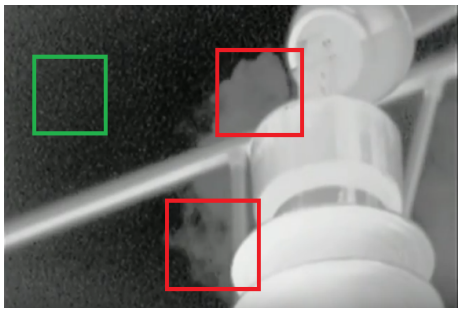

Fig. 1: An example MWIR band infrared image of VOC gas leak. Red (green) rectangles contain gas-leak regions (leak-free region). Image is downloaded from FLIR systems [12].

2. DEEP LEARNING ALGORITHMS

We utilize three kinds of deep neural networks to classify the VOC gas data. The first one is a regular convolutional neural network (CNN). The second one is an additive neural network based on addition and sign operations, and the third one is the discriminator or a Generative Adversarial Neural (GAN) network.

2.1. Convolutional Neural Network

Convolutional neural networks (or ConvNets) have been extensively used in computer vision [13, 14] and time-series data analysis [15]. In ConvNets, convolution (or correlation) between inputs and weights is used to extract local features at different scales in subsequent layers. This locality in feature extraction, along with affine transformation and non-linearity, enables the network to extract highly abstract features at high levels.

Since both the IR and chemical sensor readings are temporarily varying, we implemented a one-dimensional convolutional neural network with three convolutional layers including one dense input layer, one dense output layer and a binary classifier layer. The number of filters in the three filters are 16, 32 and 64, respectively. The size of each filter is 3 with maxpooling of size 5 being used between the layers. The output of the last maxpooling is subsequently flattened and fed to a dense layer of output size of 64, whose output is eventually fed to the binary classifier layer. We used rectified linear units (ReLU) as nonlinear activation in all hidden layers.

2.2. Additive Neural Network

Despite the great success of deep learning algorithms in time-series analysis, CNNs are computationally expensive. It may not be possible to implement a CNN in regular mobile devices. Since multiplication operations consume significant energy, a regular neural network will fail to pass an energy efficiency constraint as well.

We introduced the notion of a new "product" of two vectors as a binary operation involving sign, addition and subtraction operations, and maximum and minimum operations [16, 17]. The new class of vector products are non-Euclidean operators and the "product" of a vector with itself can be expressed in terms of the ℓ_1 norm of the vector. As a result, algorithms and methods developed using the new class of vector products will lend themselves to significantly more efficient sparse system manipulations. In this regard, let a and b be two real numbers. A "product" of a and b that we are exploring is defined initially as follows:

$$a \oplus b := \text{sgn}(a \times b)(|a| + |b|) \tag{1}$$

where the *sgn* is the *signum* function ($a = \text{sgn}(a)|a|$). The operator $\oplus$ basically performs an addition operation, but the sign of the result is the same as the sign of $a \times b$. We call the new operator the multiplication-free operator (mf-op) which can also be represented as follows:

$$a \oplus b := \text{sgn}(a)b + \text{sgn}(b)a \tag{2}$$

Let $\mathbf{x}$ and $\mathbf{y}$ be two vectors in $\mathbf{R^d}$. We define a "vector product" based on the $\oplus$ operator as follows, adapting the same notation as for the scalar case:

$$\mathbf{x} \oplus \mathbf{w} := \sum_{i=1}^{d} \text{sign}(x_i \times w_i)(|x_i| + |w_i|), \tag{3}$$

where $\mathbf{x} = [x_1, \ldots, x_d]^T, \mathbf{w} = [w_1, \ldots, w_d]^T \in \mathbf{R^d}$. This "vector product" operation, $\oplus$, requires no multiplication whatsoever. The "vector product" defined in (3) leads to a scaled version of the ℓ_1 norm: $\mathbf{x} \oplus \mathbf{x} = 2||x||_1$.

The "vector product" shown above can be used in operations that are analogous to correlation. In traditional ANNs, neurons perform inner products to compute the correlation between the input vector with the weights of the neuron. We define a new neuron by replacing the inner-product of a classical neuron by the "vector product" defined using (1) or (2). A neuron in a classical neural network is represented by the activation function

$$f(\mathbf{x}^T\mathbf{w} + b), \tag{4}$$

where $f()$ is the nonlinear activation function, $\mathbf{w} \in \mathbf{R^d}$, $b \in \mathbf{R}$ denote the weights and the bias, respectively, and $\mathbf{x} \in \mathbf{R^d}$ is the input vector. We define a new neuron by replacing the affine transform of a traditional neuron using the mf-op as follows:

$$f(\alpha(\mathbf{x} \oplus \mathbf{w}) + b), \tag{5}$$

where $\mathbf{w} \in \mathbf{R^d}$, and $\alpha, b \in \mathbf{R}$ denote the weights, the scaling coefficient and the bias, respectively. The neural network in which each neuron is represented by the activation function defined in Eqn. (4), is called Additive Neural Network (AddNet). In this work, we refer to Multi-Layer Perceptron (MLP) networks which implement the mf-operator as *MLP-AddNets*. Likewise, we refer to ConvNets which implement the mf-operator as *Conv-AddNets*. The proof of AddNet with linear and/or ReLU activation function satisfying the *universal approximation property* over the space of Lebesgue integrable functions can be found in [18]. Most neural network structures can be easily converted into an additive network structure by just replacing ordinary neurons with the activation functions defined using the mf-op. It is possible to train the AddNet using the standard back-propagation and related optimization algorithms.

Since response normalization is commonplace in deep neural network architectures [13, 19], we realized that normalization is also important in the case of AddNet, since addition results in responses of larger variances than ordinary multiplication when dealing with values smaller than unity in magnitude. In this regard, we normalize the output of the i^{th} feature map of a convolutional layer by a scalar $\alpha_i = \frac{||\mathbf{w}_i||_1}{H \times C}$ where $\mathbf{w}_i$ is the i-th filter in the filter bank, H the 1-D spatial size of the filter and C the depth of the filter (number of input channels). This normalization factor was used in the binary-weight Neural Network proposed in [20]. Thus, we perform only one multiplication per neuron instead of $H \times C$ multiplications. Alternatively,

one can use other normalization techniques, such as batch normalization [19].

Furthermore, since the signum function is not suitable for back-propagation, we redefined the derivative of the signum function in the operator to be that of the hyperbolic tangent function. Since the hyperbolic tangent function acts linearly close to zero, this makes the derivative smooth for small values.

2.3. Discriminator of GAN as Classifier

Generative Adversarial Networks (GAN) have become the benchmark in image synthesis [21, 22]. A typical GAN has a generative network and a discriminator network. In a standard application of GAN the generator network is used to synthesize artificial images from noise vectors, and the discriminator network is used to train the generator. In this paper, our aim is not to synthesize data but to make use of the adversarial nature of GAN training to obtain a discriminator network capable of classifying the input with an unbalanced set of training data. As the recordings of gas leak data may fall short of the clean air recordings for this purpose, we have the generator of the GAN compensate the data set with smaller number of data instances by producing "artificial" gas leak data during training.

In this regard, we perform a two-phase training of the GAN. First, we carry out unsupervised adversarial training of both the discriminator and the generator using the data of one of the classes and random noise. Let x^i represent the i^{th} vector of one of the classes. In this case, x^i's denote the gas leak recordings. Let z be a random noise vector, e.g. Gaussian noise or uniform noise. Let D be the discriminator and G be the generator, each of which has parameters sets θ_D and θ_G, respectively. In the adversarial stage, we seek to optimize the following loss function:

$$\max_{\theta_D} \min_{\theta_G} \sum_i log(D(x^i)) + \sum_i log(1 - D(G(z^i)) \qquad (6)$$

where $D(x^i)$ is the prediction result of the discriminator. The prediction output $D(x^i)$ should be close to 1 because x^i is "real". The generator G produces "fake" data signals from noise vector z^i, and $D(G(z^i))$ should be close to zero because $G(z^i)$ is an artificial data instance. Once training the first stage is accomplished, we move on to the second stage of supervised training of the entire training data, in which the cost function we seek to minimize is the regular binary cross entropy function CE expressed as follows:

$$CE := -\frac{1}{N}\Big(\sum_i (1 - t^i) log(1 - D(x^i)) + t^i log(D(x^i))\Big) \quad (7)$$

where $t^i \in \{0, 1\}$ denotes the true class of x^i.

When there are multiple classes we can still use the discriminator of a GAN. Let us assume that there are N-classes. In this case, the one-hot encoded label for each input is an N-dimensional vector, with all entries equal to zero, except for the k^{th} entry, where k is the true class. During training, the discriminator (or classifier) will minimize the cross-entropy of the softmax layer applied at the output layer (N logits). The generator parameters θ_G will try to attack the k^{th} output node by trying to push the sigmoidal response of that node to unity by minimizing the cross entropy CE_G of that node, which is expressed as:

$$CE_G(node\ k) = -log(D(G(z)) \qquad (8)$$

where z is a noise vector input to the generator network G of GAN. The discriminator parameters θ_D in turn will counteract the generator by trying to push the sigmoidal response of the fake examples to zero by minimizing the cross entropy CE_D given by

$$CE_D(node\ k) = -log(1 - D(G(z)) \qquad (9)$$

Notice that in Eq. 8 and Eq. 9, we consider the sigmoidal output of the logits, instead of the softmax response, as we reduce the problem into a two-class adversarial optimization. Therefore, for each data point, only one output node will be attacked at a time. In practice, since we do a mini-batch update, we take the average of the loss functions and minimize the loss functions based on the mini-batch gradients.

3. DATASETS AND EXPERIMENTAL RESULTS

3.1. Infrared VOC Dataset

Our first data set consists of infrared imaging signals of VOC gas leaks in open air and clean air recordings. Specifically, we have two classes of discrete-time signals corresponding to VOC gas leaks and clean air, respectively. Each signal is a time series containing 50 samples corresponding to two seconds of recording with a sampling rate of 25 measurements per second. The recorded value varies in open air due to background temperature variations. When there is no leak, the variance is much smaller compared to the VOC gas leak in which case the gas molecules absorb the IR light. We gathered about 30,000 VOC gas leak and 30,000 clean air data instances.

The dynamic range of the recorded signal values varies according to the background temperature and the sensors are uncalibrated in practice. Therefore, we used min-max normalization to scale signal data points between 0 and 1. The normalized signal $\hat{x}$ is obtained as follows

$$\hat{x}[n] = \frac{x[n] - min(x)}{max(x) - min(x)}, \quad n = 0, 1, \dots, 49 \qquad (10)$$

where $max(x)$ and $min(x)$ represent the maximum and minimum values of a given infrared signal x, respectively.

An infrared image of the gas leak is shown in Fig. 1. The image is obtained using an MWIR camera produced by FLIR systems [12]. VOC gas absorbs the infrared light appearing as a white cloud in the black-hot mode infrared image as shown in Fig. 1.

We used the same architecture described in Sec. 2.1 for the regular CNN and Sec. 2.2 for the AddNet. We divided our data set into three disjoint sets. The training data consists of 8,000 recordings of each class. Another set of 8,000 recordings of each class are used as the validation data set. The rest of the data was reserved for testing. We trained our networks using the RMSProp optimizer algorithm [23]. We tested whether using dropout helps achieve better results [24]. We used a dropout rate of 50%. As for the GAN approach, our generator is a MLP with one hidden layer of size 256. The regular convolutional neural network and AddNet exhibit comparable results. We obtained an accuracy of 99.8% for no-gas data and 99.7% for gas-leak data for a regular ConvNet. AddNet achieved a recognition rate of 98.9% for no-gas data and 99.3% for gas-leak data.

In the second set of experiments, we assumed that we have an unbalanced data set. In practice, we may not have VOC or ammonia gas leak recordings as clean air. We trained the models with only 50 recordings of gas leak signals against 8,000 recordings of clean air recordings. The test data set contains 14,000 recording instances of VOC gas leaks and clean air recordings. Classification results are also summarized in Table 1. AddNet produces the best results but the discriminator of the GAN Network is also quite close to AddNet.

Model	No-gas Accuracy	Gas-leak Accuracy	Total Accuracy
ConvNet (dropout 50%)	98.3%	95.8%	97.1%
ConvNet no dropout	98.0%	94.2%	96.1%
AddNet (dropout 50%)	98.2%	96.0%	97.1%
AddNet (no dropout)	99.1%	97.3%	98.2%
Discriminator of GAN	99.0%	97.1%	98.1%

Table 1: Accuracy results for infrared VOC data. Classifiers are trained with only 50 VOC gas leak recordings vs 8000 clean air recordings.

We also investigated pruning the weights in both AddNet and ConvNet during inference. In this regard, we discard the magnitudes of the smallest weights in magnitude while keeping their sign information. We keep the bias coefficients and the coefficients of the last layer intact. Results of different pruning rates are given in Table 2. Apparently, in AddNet, we can discard the magnitude information of the weights up to a high rate (67.4%) without severely degrading performance. On the other hand, the magnitude information is quite critical in the case of a regular ConvNet. These results clearly show the advantages of AddNet, which not only occupies less space in the memory of a mobile device but also consumes less energy because it performs much fewer arithmetical operations during inference.

Model accuracy	Weight Compression Rate (smallest K%)					
	0	16.1	19.7	67.4	76.8	86.6
AddNet	98.9	97.2	97.9	98.0	97.1	61.3
ConvNet	99.8	67.4	–	–	–	–

Table 2: Effect of compressing weights of both AddNet and ConvNet by discarding the magnitude of the smallest $K\%$ while keeping the sign information. ConvNet cannot produce any reasonable results after 16.1% compression rate. The compression rate is estimated by allocating 32 bits to intact weight values and 1 bit for every binarized weight factor.

3.2. Chemical Gas Sensor Array Drift Dataset

The second set of experiments is carried out using the publicly available chemical VOC gas sensor drift data set obtained by Vergara *et al.* [8]. The data set was obtained by exposing an array of 16 different chemical sensors to 6 different types of gas mixtures (ammonia, acetone, ethylene, ethanol, toluene and acetaldehyde) at different concentration levels. Each data record is a vector time series. Vectors contain 8 feature parameters extracted from the sensor time series signals during a gas release experiment, conducted over a period of 3 years at UCSD. The feature parameters include maximal resistance change and its normalized value during an experiment. Other 6 parameters are maxima and minima of exponential moving average (ema_α) transform values with α set to 0.1, 0.01 and 0.001. Since there are 16 sensors, a total of $16 \times 8 = 128$ feature values are recorded per experiment. The data set is divided into 10 batches ordered chronologically. Full details about the experiment and the data set can be found in [8].

We carried out our classification tasks by training our neural networks for $N = 5$ batches and test on the successive batches. This is identical to the sensor drift estimation approach given in [8]. As feature values have huge variances, we opted to use the signed square root scalar-valued normalization of every feature value. We trained a multi-layer perceptron (MLP) of two hidden layers, each of which has 512 output units, and an output layer. We trained the network for 100 epochs using RMSProp optimizer [23]. We applied a dropout of rate 20% and used a batch size of 128. In order to augment the data, we added zero-mean Gaussian noise with standard deviation of 0.1.

A numerical comparison of the proposed methods to the SVM-classifier ensemble used in [8] is given in Table 3. In general, the AddNet-MLP, the MLP and the multi-class GAN discriminator produce better sensor drift estimation results than the SVM based method.

Batch ID	SVM Classifier Ensemble [8]	MLP	AddNet-MLP	Disc. of GAN
Batch 3	87.8	**98.6**	**98.6**	98.3
Batch 4	**90.6**	83.8	75.1	71.4
Batch 5	72.1	**99.5**	99.4	98.4
Batch 6	44.5	74.9	**75.9**	72.3
Batch 7	42.5	59.8	57.4	**61.5**
Batch 8	29.9	34.0	34.0	**62.3**
Batch 9	59.8	31.6	38.9	**63.2**
Batch 10	39.7	47.3	**54.3**	43.8

Table 3: Comparative accuracy (in %) results between the different models when training on batches 1 and 2 and testing on batches 3-10

4. CONCLUSIONS

In this paper we described a cyber-physical system (CPS) for detecting VOC gas leaks. The CPS is comprised of ordinary people or first responders using small, low-cost sensors attached to their smart phones or mobile devices. One major challenge is the sensor drift problem in chemical sensors. Similarly, passive IR sensors also suffer from infrared light reflections from the background in open air. Therefore, thresholding the sensor outputs for VOC gas leak is not a reliable option. To address the problem of sensor drift, we analyzed the time-varying signal waveforms that sensors generate using neural networks. In addition to a regular deep neural network we use the AddNet and the discriminator of a GAN as a classifier. AddNet produces comparable results to a regular deep neural network without the need to perform vector multiplication operations, which require energy consuming GPU processing. Furthermore, the weights of AddNet are highly compressible, with no resultant degradation in performance, suggesting the possibility of AddNet being used in mobile devices forming such CPS, to deliver accuracy and frugality at the same time. While it can be observed that the discriminator of GAN also performs as well as the other two neural networks in the two classification problems studied in this paper, it unfortunately suffers from inordinate computational cost.

5. REFERENCES

[1] Andy P Jones, "Indoor air quality and health," *Atmospheric environment*, vol. 33, no. 28, pp. 4535–4564, 1999.

[2] SK Brown, Malcolm R Sim, Michael J Abramson, and Christopher N Gray, "Concentrations of volatile organic compounds in indoor air–a review," *Indoor air*, vol. 4, no. 2, pp. 123–134, 1994.

[3] Cesare Maltoni, Adriano Ciliberti, Giuliano Cotti, Barbara Conti, and Fiorella Belpoggi, "Benzene, an experimental multipotential carcinogen: results of the long-term bioassays performed at the Bologna Institute of Oncology.," *Environmental health perspectives*, vol. 82, pp. 109, 1989.

[4] Jian Kang, Junjie Liu, and Jingjing Pei, "The indoor volatile organic compound (VOC) characteristics and source identification in a new university campus in Tianjin, China," *Journal of the Air & Waste Management Association*, vol. 67, no. 6, pp. 725–737, 2017.

[5] Wolfgang Göpel and Klaus-Dieter Schierbaum, "Definitions and typical examples," *Sensors: Chemical and Biochemical Sensors-Part I*, vol. 2, pp. 1–27, 1991.

[6] F Davide, C Di Natale, M Holmberg, and F Winquist, "Frequency analysis of drift in chemical sensors," in *Proceedings of the 1st Italian Conference on Sensors and Microsystems, Rome, Italy*, 1996, pp. 150–154.

[7] Marzia Zuppa, Cosimo Distante, Pietro Siciliano, and Krishna C Persaud, "Drift counteraction with multiple self-organising maps for an electronic nose," *Sensors and Actuators B: Chemical*, vol. 98, no. 2-3, pp. 305–317, 2004.

[8] Alexander Vergara, Shankar Vembu, Tuba Ayhan, Margaret A Ryan, Margie L Homer, and Ramón Huerta, "Chemical gas sensor drift compensation using classifier ensembles," *Sensors and Actuators B: Chemical*, vol. 166, pp. 320–329, 2012.

[9] Tom Artursson, Tomas Eklöv, Ingemar Lundström, Per Mårtensson, Michael Sjöström, and Martin Holmberg, "Drift correction for gas sensors using multivariate methods," *Journal of chemometrics*, vol. 14, no. 5-6, pp. 711–723, 2000.

[10] Fatih Erden, E Birey Soyer, B Ugur Toreyin, and A Enis Cetin, "VOC gas leak detection using pyro-electric infrared sensors," in *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*. IEEE, 2010, pp. 1682–1685.

[11] Ahmet Enis Cetin and Behcet Ugur Toreyin, "Method, device and system for determining the presence of volatile organic compounds (VOC) in video," Apr. 30 2013, US Patent 8,432,451.

[12] Ferret.com.au, "FLIR systems gas find infrared camera demonstration," Sep 2012.

[13] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[14] Yann LeCun, Yoshua Bengio, et al., "Convolutional networks for images, speech, and time series," *The handbook of brain theory and neural networks*, vol. 3361, no. 10, 1995.

[15] Martin Längkvist, Lars Karlsson, and Amy Loutfi, "A review of unsupervised feature learning and deep learning for time-series modeling," *Pattern Recognition Letters*, vol. 42, pp. 11–24, 2014.

[16] Diaa Badawi, Ece Akhan, Ma'en Mallah, Ayşegül Üner, Rengül Çetin-Atalay, and A. Enis Çetin, "Multiplication free neural network for cancer stem cell detection in h-and-e stained liver images," in *Compressive Sensing VI: From Diverse Modalities to Big Data Analytics*. International Society for Optics and Photonics, 2017, vol. 10211, p. 102110C.

[17] Arman Afrasiyabi, Diaa Badawi, Bariş Nasir, Ozan Yildi, Fatoş T Yarman Vural, and A Enis Çetin, "Non-euclidean vector product for neural networks," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 6862–6866.

[18] George Cybenko, "Approximation by superpositions of a sigmoidal function," *Mathematics of control, signals and systems*, vol. 2, no. 4, pp. 303–314, 1989.

[19] Sergey Ioffe and Christian Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.

[20] Mohammad Rastegari, Vicente Ordonez, Joseph Redmon, and Ali Farhadi, "XNOR-Net: ImagenNet classification using binary convolutional neural networks," in *European Conference on Computer Vision*. Springer, 2016, pp. 525–542.

[21] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[22] Alec Radford, Luke Metz, and Soumith Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *arXiv preprint arXiv:1511.06434*, 2015.

[23] Tijmen Tieleman and Geoffrey Hinton, "Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude," *COURSERA: Neural networks for machine learning*, vol. 4, no. 2, pp. 26–31, 2012.

[24] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.