9th International Conference on Information Science and Technology (ICIST), Hulunbuir, Inner Mongolia, China; August 2-5, 2019

# Data Augmentation for Mixed Spectral Signatures Coupled with Convolutional Neural Networks

[1]Juan F. Ramirez Rochac and [2]Nian Zhang
[1]Dept. of Computer Science & Information Technology, [2]Dept. of Electrical and Computer Engineering
University of the District of Columbia
Washington, D.C., USA 20008
Email: jrochac@udc.edu, nzhang@udc.edu

[3]Jiang Xiong and [4]Jing Zhong
College of Computer Science and Engineering
Chongqing Three Gorges University
Chongqing, China 404000
Email: xjcq123@sohu.com, zhongandy@sohu.com

[5]Timothy Oladunni
Dept. of Computer Science & Information Technology
University of the District of Columbia
Washington, D.C., USA 20008
Email: timothy.oladunni@udc.edu

***Abstract*—Advances in photo-thermal infrared imaging spectroscopy technologies have come a long way. These advances include the development of quantum cascade laser (QCL) which greatly improves the measurement of infrared absorption. Despite the progress made, a few fundamental limitations of infrared spectroscopy have prevented its processing in deep learning models. Direct application of convolutional neural networks becomes difficult, because of the high dimensionality, and low number of samples. To overcome these limitations, in this paper, we explore the applicability of data augmentation techniques to photo-thermal infrared imaging data. First, we establish a baseline using the raw data and a convolutional neural network (CNN). Second, we apply principal component analysis (PCA) and proceed to applying CNN. Third, we employ different data augmentation techniques before applying CNN. After that, we estimate the performance of our models by using k-fold cross-validation and calculating the confusion matrix and classification accuracy. Our GDAC+CNN model achieved an average of 99% accuracy.**

***Keywords*— deep learning; convolutional neural networks; photo-thermal infrared; principal component analysis; data augmentation; multiclass classification**

## I. INTRODUCTION

Deep learning featured by deep structured learning is a technique that enables a system to automatically discover the features needed for feature detection or classification from raw data. A deep neural network (DNN) is a type of artificial neural network (ANN) with deep structured layers between the input and output layers. DNNs have the ability to model complex non-linear relationships. The network will be trained to express an object as a layered composition of primitives. The extra layers enable composition of features from lower layers, potentially modeling complex data with fewer units than a similarly performing shallow network. In deep learning, a convolutional neural network (CNN) is a class of deep neural networks, which has provided evidence of superior performance in the applications of image classification, image and video recognition, and medical image processing, analysis and visualization. Data augmentation has shown promising results, especially coupled with deep learning models in achieving higher degree of accuracy of image classification tasks [1]. It has provided a solution to small datasets by generating new, derived samples using different mathematical transformations, such as affine transformations and Pixel-wise transformations. However, affine transformations cannot be directly applied to photo-thermal infrared images [2], because any flipping, rotating, or shifting of the original image would result in a different signature that corresponds to a different material. Pixel-wise transformations present an option to data augmentation for photo-thermal infrared images, because a change in the level of noise, blur and contrast will not directly impact the signature of the image. Yet unbound, arbitrary pixel-wise changes may distort the underlining image to the point of chaos.

In this paper, we study how different data augmentation techniques impacts for the performance of a CNN for photo-thermal infrared images. We also propose a new data augmentation technique that integrates signal-to-noise ratio (SNR) with an additive white Gaussian noise (AWGN) to generate derived data samples suited for multiclass classification in convolutional neural networks. The rest of this paper is organized as follows. In Section 2, the photo-thermal infrared imaging spectroscope technique and the CNN architecture are introduced. In Section 3, the methodology is described, from principal component analysis (PCA) to data augmentation (DA), from convolutional neural networks (CNN) to our proposed Gaussian data augmentation (GDA) technique. In Section 4, the experimental results are presented. In Section 5, the conclusions are given.

## II. BACKGROUND

### A. Photo-thermal IR Imaging Spectroscopy Data Set

In the photo-thermal IR imaging spectroscopy (PT-IRIS) approach, a desired number of infrared (IR) quantum cascade lasers are adapted to strong absorption bands in the analytes while an IR focal plane array is used to image the reflecting thermal radiations. An IR laser instantly illuminates the surface that may be contaminated by targeted analytes. The total PT-IRIS signal is approximated as a linear combination of the particle signal and substrate signal [3]. The temporal behavior of the signal indicates the thermal properties of the substrate [4]. Spatial (pixel to pixel) variations in the PT-IRIS

images can help distinguish particles from substrate [5]. The feature vectors for the PT-IRIS signals can be retrieved from the temperature increase during each laser pulse as a normalized power of the laser pulse [6].

### B. CNN Architecture

The CNN is usually designed to imitate human visual processing. Compared with fully connected neural networks, CNN has fewer parameters and is easier to escape from vanishing gradients problem [7]. The CNN contains three different layers: convolution layer, pooling layer, and fully connected layer. Convolution layer is the most important layer of a CNN. Such architecture makes network concentrate on low-level features in the previous layer and assemble low-level feature into high-level features in the latter layers [8]. The goal of pooling layers is to subsample the feature maps. This will reduce the computation burden. Convolution and pooling layers are used for feature exaction. Fully connect layer is used for classification like the situation in multilayer perceptron.

## III. Methodology

The methodology is divided into four parts. We first review the principal component analysis technique. Second, we describe data augmentation techniques that use affine transformations, such as, flipping, rotating, and shifting as well as apply pixel-wise transformations, such as, blur, noise, and contrast. Third, we describe our proposed a Gaussian data augmentation (GDA). And fourth, we define a convolutional neural network for multiclass classification.

### A. Principal Component Analysis

Principal Component Analysis is among the most commonly used approach for dimensionality reduction and data representation [9] and it has been widely used to translate highly-dimensional data into a new feature space with lower dimensionality in a vast range of fields. The goal of this translation is to reduce the number of dimensions (or feature vectors) while maintaining all relevant information and variability of the data [10]. Another great benefit is that PCA's transformation also removes anomalies and outliers from the raw data [11].

### B. Data Augmentation

Data augmentation has shown promising results, especially coupled with deep learning models in achieving higher degree of accuracy of image classification tasks [1]. It also provides a solution to small datasets by generating new, derived samples using different mathematical transformations. Some of these transformations consist of using a combination of mathematical equations or kernels to modify the original images [12]. For instance, affine transformations generate a derived image by flipping, rotating or shifting the original image. Pixel-wise transformations produce derived images by adding noise, increasing the contrast or/and blur on the original image. All original and derived images are fed into the classification model. Basic transformations are depicted in Fig. 1. On the top left corner with have the original image. The other eight images in this figure are generated by different data augmentation techniques.

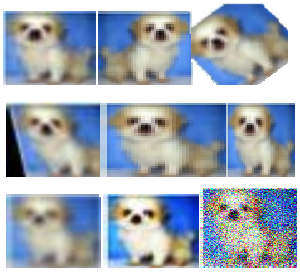


Fig 1. Data Augmentation Transformations

### C. Affine Transformations

Affine transformations also called linear transformations are mathematically mapping methods that are used in image processing to correct deformations. Translation, rotation, isotropic scaling and shearing are among the most common affine transformations. Fig. 1 depicts, on the top and middle rows, how these transformations will alter a regular image.

### D. Pixel-wise Transformations

Pixel-wise transformations are based on the idea of transforming each pixel in the image using a point or pixel operator. Point operators produces each new output pixel value based on its corresponding input pixel value. Some of these pixel-wise transformations use global/local parameters while others are based on exclusively on the input pixel value. For example, the *blurring* effect can be achieved by convoluting the original image **Img** with a moving average denoted by a kernel **K**, as shown in Equation 1.

$$\textit{Blurry Image} = \mathbf{Img} * \mathbf{K} \tag{1}$$

### E. Gaussian Data Augmentation

Our proposed data augmentation technique uses an Additive white Gaussian noise (AWGN) to produce derived images. The motivation for this technique comes from information theory and signal processing and its applications to digital communication systems. In communication system, the received signal is equal to the sent or original signal plus the effect of the channel [13] as shown in Equation 2. Both the received signal $Y(t)$ and the original signal $X(t)$ are considered random variables and the effect of the channel $N(t)$ follows a Gaussian distribution with a probability density function presented in Equation 3. Our random noise will follow a Gaussian Normal distribution $N(\mu,\sigma)$ where $\mu$ is zero and $\sigma$ is determined from the signal-to-noise ratio (SNR) formula presented in Equation 4. We target a SNR of 60 dB to obtain an AWGN that is not apparent.

$$Y(t) = X(t) + N(t) \tag{2}$$

$$f(x|\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{(x-\mu)^2}{2\sigma^2}\right] \tag{3}$$

$$SNR_{dB} = 20 \cdot \log_{10} \frac{\mu_{signal}}{\sigma_{noise}} \tag{4}$$

### F. Convolutional Neural Networks

Deep machine learning is a model that learns to perform classification tasks directly from the data. Convolutional Neural Networks (CNN) is one type of deep learning model. The term "deep" refers to the number of layers.

Fig. 2 depicts the layers of our CNN. All experiments were conducted using the model presented in Fig. 2. Input data will be resized and augmented before adding to the network. The actions of data augmentation can be randomly flipping the pictures and randomly cropping the pictures, which force CNN to learn more essential feature.

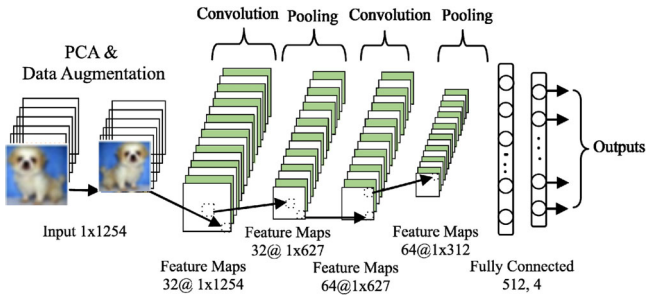



Fig 2. CNN Model Overview

Fig. 2 depicts the layers of our CNN. All experiments were conducted using the model presented in Fig. 2. Input data will be resized and augmented before adding to the network. The actions of data augmentation can be randomly flipping the pictures and randomly cropping the pictures, which force CNN to learn more essential feature.

TABLE I. CNN PARAMETERS IN SIMULATION

| Hyper-parameter | Value |
|---|---|
| Filter of 1st Convolution Layer | [2, 2, 32, 32] |
| Filter of 2nd Convolution Layer | [2, 2, 64, 64] |
| Strides of Convolution Layer | [3, 3] |
| Padding | same |
| Activation | ReLU |
| Strides of Pooling Layer | [2, 2] |
| Fully Connected Layer Size | 3104×12352× 4 |

In the simulation, the hyper-parameters of the CNN is shown in Table I. The numbers of convolution layer filter parameters stand for [filter height, filter width, input channels, output channels], respectively [14]. The activation for all layers is ReLU. The strides of the two pooling layers are the same. The CNN structure has three fully connected layers, the number of neurons in these three layers is [3104, 12352, *flatten*, 4]. The value of *flatten* varies for CNN and PCA+CNN because it is derived from the input shape, that is the number of input features or principal components. Implementation was done using Keras [15] and TensorFlow. Runs consisted of 20 Epochs with batch size of 10. Runtime platform was a Tesla k80 GPU, with 2496 CUDA cores, 12GB GDDR5 VRAM.

### G. *Performacne Metrics*

The confusion matrix was computed for each data augmentation technique to compare overall accuracy. Fig. 3 is a representation of the data partition for a five-fold cross-validation, where 20% of the entire dataset is used for testing and 80% is used for training. To help reduce the effects of overfitting, a different 20% is used for testing for each fold/iteration in a round robin fashion. For Deep Learning training, datasets are divided into 3 parts: training (60%), validation (20%) and test (20%). Training and validation sets are used during the fitness phase while only test samples are used in the prediction phase.

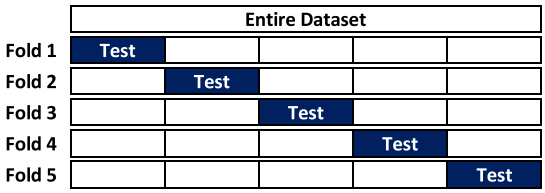



Fig 3. Five-fold Cross-Validation Partitions

## IV. EXPERIMENTAL RESULTS

We start with the results of our baseline, namely, CNN and PCA+CNN. Then we present a comparison between different data augmentation techniques. Three affine transformations and three pixel-wise transformations are evaluated. We then end with a presentation of the results of our Gaussian Data Augmentation technique GDA+CNN for different multipliers. The main goal of our experiments is to comprehensively measure multiclass classification accuracy for different data augmentation techniques. To this end, 20-fold cross-validation was performed for each data augmentation technique.

### A. *Convolutional Neural Network (CNN and PCA+CNN)*

In this subsection, a baseline is established for CNN with and without PCA. Fig. 4 shows the accuracy (left column) and loss (right column) as functions of the number of epochs for training samples in blue and validation samples in green.

Table II presents the testing averages for Precision, Recall, F1-score and Support. CNN and PCA+CNN were evaluated using 20-fold cross-validation.

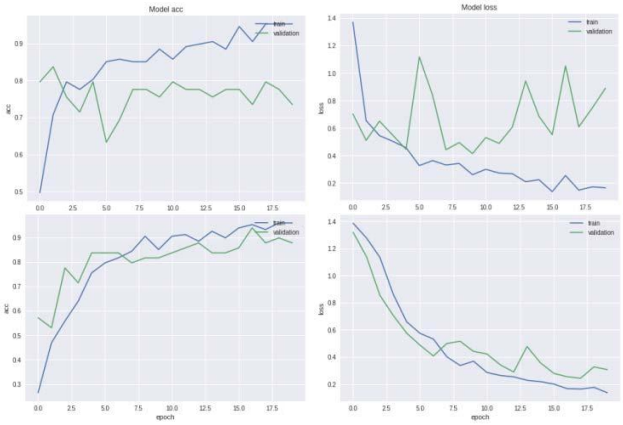



Fig 4. Accuracy & Loss of CNN without preprocessing (top row), and PCA+CNN (bottom row)

TABLE II. CLASSIFCATION REPORT FOR CNN

| Classes | Precision | Recall | F1-score | Support |
|---|---|---|---|---|
| **Convolutional Neural Network** | | | | |
| **AN** | 1.00 | 0.57 | 0.73 | 7 |
| **RDX** | 0.44 | 1.00 | 0.62 | 4 |
| **Sucrose** | 1.00 | 0.83 | 0.91 | 6 |
| **TNT** | 1.00 | 0.88 | 0.93 | 8 |
| **Principal Component** | | | | |
| **AN** | 1.00 | 0.67 | 0.80 | 6 |
| **RDX** | 0.78 | 0.88 | 0.82 | 8 |
| **Sucrose** | 1.00 | 1.00 | 1.00 | 8 |
| **TNT** | 0.50 | 0.67 | 0.57 | 3 |

Fig. 5 presents the Confusion Matrix for CNN and PCA+CNN. Ground truth is on the Y-axis and the predicted labels on the X-axis. Lighter shades of green represent less number of samples accurately detected. Hence, a darker main diagonal means a better prediction.

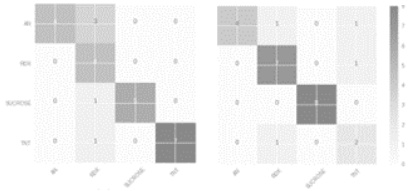


Fig 5. Confusion Matrix for CNN (left), and PCA+CNN (right)

## B. *DA+CNN using Affine Transforamtions*

In this subsection, Data Augmentation is evaluated for three different Affine Transformations. Flipping, rotation and translation are used to derive additional samples. Fig. 6 shows the accuracy and loss over the number of epochs for each transformation.

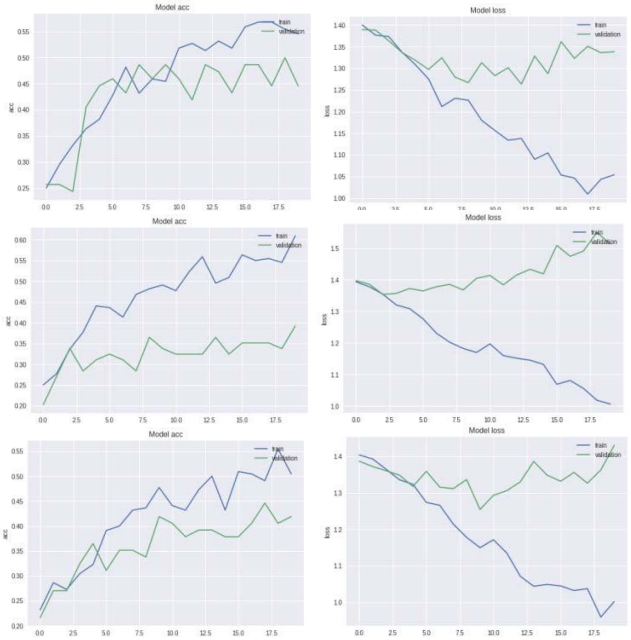



Fig 6. Accuracy & Loss of DA+CNN using Flipping (top), Translation (center) and Rotation (bottom)

Table III presents the testing averages for Precision, Recall, F1-score and Support. Three variants of DA+CNN were evaluated, flipping, rotation and translation. Results were compiled from our cross-validation.

TABLE III. CLASSIFCATION REPORT FOR DA+CNN USING AFFINE TRANSFORMATION

| Classes | Precision | Recall | F1-score | Support |
|---|---|---|---|---|
| **Flipping transformation** | | | | |
| **AN** | 0.00 | 0.00 | 0.00 | 5 |
| **RDX** | 1.00 | 0.25 | 0.40 | 4 |
| **Sucrose** | 0.75 | 0.43 | 0.55 | 7 |
| **TNT** | 1.00 | 0.44 | 0.62 | 9 |
| **Translation or Shifting transformation** | | | | |
| **AN** | 1.00 | 0.56 | 0.71 | 9 |
| **RDX** | 1.00 | 0.50 | 0.67 | 4 |
| **Sucrose** | 1.00 | 0.33 | 0.50 | 6 |
| **TNT** | 1.00 | 0.67 | 0.80 | 6 |
| **Rotating transformation** | | | | |
| **AN** | 0.42 | 1.00 | 0.59 | 5 |
| **RDX** | 1.00 | 0.60 | 0.75 | 5 |
| **Sucrose** | 1.00 | 0.43 | 0.60 | 7 |
| **TNT** | 0.80 | 0.50 | 0.62 | 8 |

Fig. 7 presents the corresponding Confusion Matrix for DA+CNN using flipping, translation and rotation as its affine transformations of choice. Ground truth is on the vertical axis and the predicted labels are on the horizontal axis. Note that lighter shades of red represent fewer number of samples and darker shades of red represent larger number of samples. Hence, a darker diagonal means a better prediction. The darker red regions under the main diagonal indicate that presence of false positive while the darker regions above the main diagonal indicate the presence of false negatives.

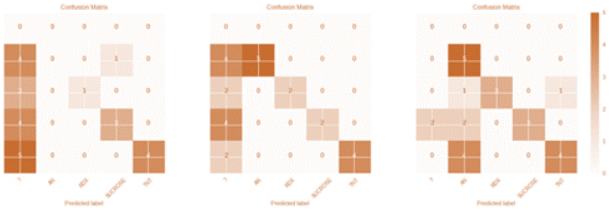
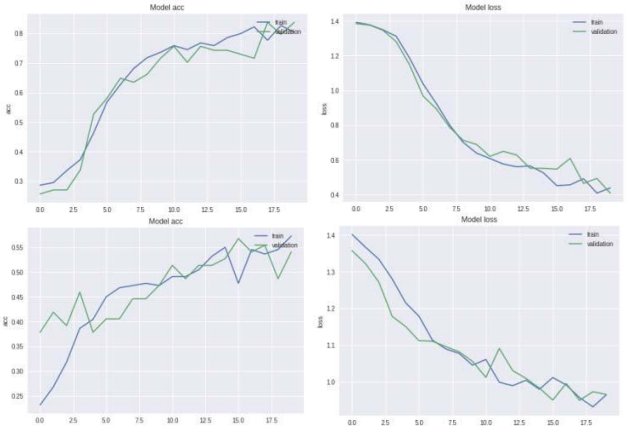


Fig 7. Confusion Matrix for DA+CNN using Flipping (left), Translation (middle) and Rotation (right)

From the Confusion Matrix, it is clear that the model was not able to determine the class for a significant number of samples. All three affine transformations decrease the overall accuracy of our CNN model. For instance, flipping decreases the model's accuracy to 30% on test samples.

## C. *DA+CNN using Pixel-wise Transforamtions*

In this subsection, Data Augmentation is evaluated for three pixel-wise transformations. Contrast, Noise and Blur are used to derive additional samples and in conjunction with the original samples our CNN model is trained and validated.



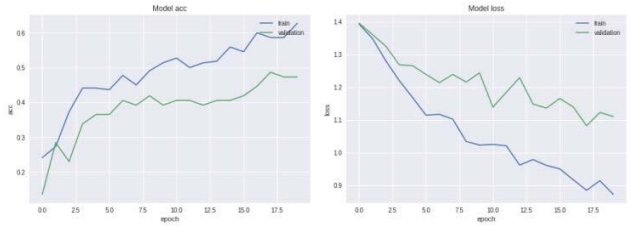


Fig 8. Accuracy & Loss of DA+CNN using Contrast (top), Noise (center) and Blur (bottom)

Fig. 8 shows the accuracy (left column) and loss (right column) as functions of epochs for the training set in blue and validation set in green.

Table IV presents the testing averages for Precision, Recall, F1-score and Support. Using only pixel-wise transformations, DA+CNN was evaluated. Results indicate that pixel-wise transformations are suitable for our type of data. Furthermore, results from a 20-fold cross-validation experimental setting suggest that noise is the better option in term of the weighted average for precision and recall.

TABLE IV. CLASSIFCATION REPORT FOR DA+CNN USING PIXEL-WISE TRANSFORMATIONS

| Classes | Precision | Recall | F1-score | Support |
|---|---|---|---|---|
| **Contrast variation** | | | | |
| **AN** | 0.75 | 1.00 | 0.86 | 6 |
| **RDX** | 1.00 | 0.44 | 0.62 | 9 |
| **Sucrose** | 1.00 | 0.75 | 0.80 | 8 |
| **TNT** | 1.00 | 0.50 | 0.67 | 2 |
| **Noise variation** | | | | |
| **AN** | 0.86 | 0.86 | 0.86 | 7 |
| **RDX** | 1.00 | 0.75 | 0.86 | 8 |
| **Sucrose** | 0.67 | 0.80 | 0.73 | 5 |
| **TNT** | 1.00 | 1.00 | 1.00 | 5 |
| **Blur variation** | | | | |
| **AN** | 0.57 | 1.00 | 0.73 | 4 |
| **RDX** | 1.00 | 0.80 | 0.89 | 5 |
| **Sucrose** | 1.00 | 0.62 | 0.77 | 8 |
| **TNT** | 1.00 | 0.88 | 0.93 | 8 |

Fig. 9 presents the corresponding Confusion Matrix for DA+CNN using pixel-wise transformations. Again, ground truth is on the vertical axis and predicted labels are on the horizontal axis. Note that lighter shades of purple represent fewer number of samples and darker shades of purple represent larger number of samples. The presence of darker regions under or above the main diagonal means a poor prediction.

In the Confusion Matrix comparison in Fig. 9, there are no large darker regions outside the main diagonal; only soft shades of purple are visible under the main diagonal.

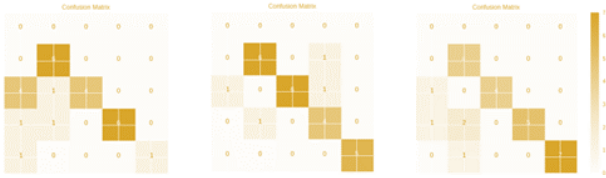


Fig 9. Confusion Matrix for DA+CNN using Contrast (left), Noise (middle) and Blur (right)

This suggests that pixel-wise transformations achieve average levels of accuracy but with a number of false positives predictions. One crucial point is that by using noise our CNN model is able to predict with a very promising level of accuracy. In our experiments the weighted average for precision and recall on test samples was of 84%. Therefore, noise was further explored and evaluated.

### *D. GDA+CNN using different multipliers*

This subsection explores and evaluates Gaussian noise as a Data augmentation technique. In particular, additive white Gaussian noise (AWGN) is tested with different multiplier. Fig. 10 shows the model's accuracy and loss, left and right columns correspondingly, as functions of the number of epochs for training and validation sets. The top row depicts the results when a multiplier $m$=1 is used; the middle row $m$=10, and in the bottom row $m$=100.

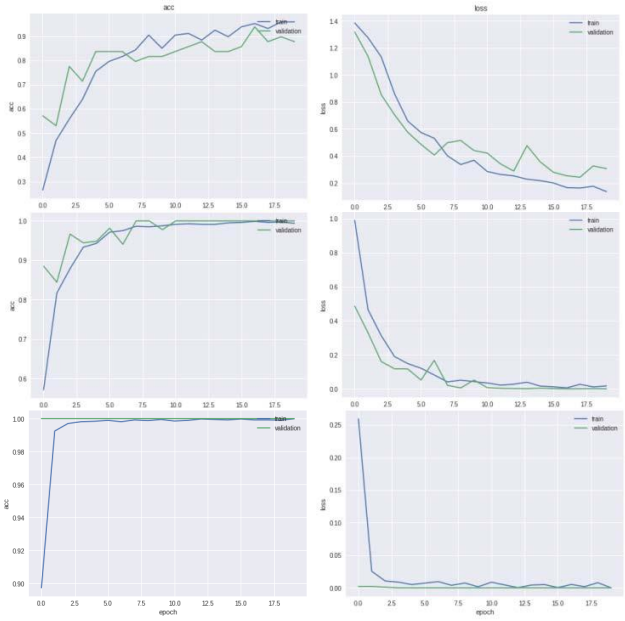


Fig 10. Accuracy & Loss of GDAI+CNN (top row) GDAX+CNN (center row) and GDAC+CNN (bottom row)

Table V presents the testing averages for Precision, Recall, F1-score and Support for three variants of our GDA+CNN approach. Results were compiled from our 20-fold cross-validation experiments.

TABLE V. CLASSIFCATION REPORT FOR GDA+CNN

| Classes | Precision | Recall | F1-score | Support |
|---|---|---|---|---|
| **GDAI+CNN (~84% Accuracy)** | | | | |
| AN | 1.00 | 0.67 | 0.80 | 6 |
| RDX | 0.78 | 0.88 | 0.82 | 8 |
| Sucrose | 1.00 | 1.00 | 1.00 | 8 |
| TNT | 0.50 | 0.67 | 0.57 | 3 |
| **GDAX+CNN (~96% Accuracy)** | | | | |
| AN | 1.00 | 1.00 | 1.00 | 6 |
| RDX | 0.86 | 1.00 | 0.92 | 6 |
| Sucrose | 1.00 | 1.00 | 1.00 | 5 |
| TNT | 1.00 | 0.88 | 0.93 | 8 |
| **GDAC+CNN (~99% Accuracy)** | | | | |
| AN | 1.00 | 1.00 | 1.00 | 5 |

| | | | | |
|---|---|---|---|---|
| RDX | 1.00 | 1.00 | 1.00 | 6 |
| Sucrose | 1.00 | 1.00 | 1.00 | 6 |
| TNT | 1.00 | 1.00 | 1.00 | 8 |

Fig. 11 presents a comparison between different variant of our Gaussian Data Augmentation, namely, GDAI+CNN which uses a multiplier of m=1, GDAX+CNN which uses a multiplier of m=10 and GDAC+CNN which uses a multiplier of m=100. As before, ground truth is placed on the vertical axis and the predicted labels are placed on the horizontal axis. Note there almost no shades of blue under or above the main diagonal. This indicates that classification accuracy is very high. Our GDAC+CNN variant achieve 99% accuracy on test data. Cross-validation was applied to rule out overfitting. In our experiments the weighted average for precision and recall on test samples went from 84% for GDAI to 96% for GDAX to 99% for GDAC.

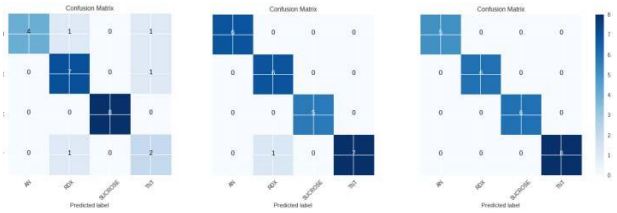


Fig. 11. Confusion Matrix for GDAI+CNN (left), GDAX+CNN (middle) and GDAC+CNN (right)

Table VI presents the consolidated results of our experiments. In our multiclass classification problem, the goal is to maximize accuracy using limited training samples. Accuracy was averaged and the standard deviation was used to measure relative error. The results of the experiments performed on the PT-IRIS data suggest that our Gaussian Data Augmentation technique provides a potential solution to the problem of limited labeled samples.

TABLE VI. ACCURACY COMPARISON SUMMARY

| Technique | Accuracy (%) | | |
|---|---|---|---|
| Raw CNN | 80.7 | ± | 9.1 |
| PCA+CNN | 75.2 | ± | 11 |
| Flip+CNN | 29.4 | ± | 12 |
| Rotation+CNN | 66.2 | ± | 8.8 |
| Translation+CNN | 50.4 | ± | 15 |
| Blur+CNN | 80.0 | ± | 10 |
| Noise+CNN | 83.4 | ± | 8.1 |
| Contrast+CNN | 67.8 | ± | 10 |
| GDAI+CNN | 84.0 | ± | 6.5 |
| GDAX+CNN | 96.4 | ± | 5.0 |
| GDAC+CNN | 99.0 | ± | 2.9 |

## CONCLUSION

Different data augmentation techniques in deep learning using a convolutional neural network were explored on a photo-thermal infrared imaging dataset. PCA and DA methods were used to preprocess multiclass classification input and predict with a high degree of accuracy. Initially, a three layer CNN was designed and performance was measured in terms of accuracy. Then PCA was applied to the dataset as a preprocessing step to the CNN model. CNN and PCA+CNN were used as baselines. Then DA was also applied as a second preprocessing step using different affine and pixel-wise transformations. Our Gaussian DA (GDA) was evaluated for different multipliers. For PCA+CNN, the overall performance decreased by 5% accuracy as compared to CNN alone. For DA+CNN using affine transformations, the overall performance was in average 50% accuracy. For DA+CNN using pixel-wise transformations, the performance was in average 77% accuracy. In this set of techniques, adding random noise improved the classification accuracy by 4% as compared to alone. Furthermore, when evaluating GDA+CNN, the overall accuracy increased to 99% accuracy.

Our results suggest that GDA+CNN is the better approach to multiclass classification of the PT-IRIS data. In the future, GDA should be further explored on different types of datasets, including CT scans, MRI images, and remote sensing datasets.

## ACKNOWLEDGMENT

This work was supported in part by the National Science Foundation (NSF) grants, HRD #1505509, HRD #1533479, and DUE #1654474.

## REFERENCES

[1] E. Jannik Bjerrum, M.Glahder, T.Skov, “Data augmentation of spectral data for convolutional neural network (CNN) based deep chemometrics,” Cornell University, arXiv.org, 2017.

[2] J. Wang, L. Perez, “The effectiveness of data augmentation in image classification using deep learning,” Cornell University, arXiv.org, 2017.

[3] N. Zhang and K. Leatham, “Feature Selection Based on SVM in Photo-Thermal Infrared (IR) Imaging Spectroscopy Classification with Limited Training Samples,” *WSEAS Transactions on Signal Processing,* vol. 13, no. 33, pp. 285-292, 2017.

[4] J. F. Ramirez Rochac, N. Zhang, and P. Behera, “Design of adaptive feature extraction algorithm based on fuzzy classifier in hyperspectral imagery classification for big data analysis,” *The 12th World Congress on Intelligent Control and Automation (WCICA 2016)*, Guilin, China, pp. 1046 – 1051, 2016.

[5] J. F. Ramirez Rochac and N. Zhang, “Reference clusters based feature extraction approach for mixed spectral signatures with dimensionality disparity,” *10th Annual IEEE International Systems Conference (IEEE SysCon 2016)*, Orlando, Florida, pp. 1 – 5, 2016.

[6] J. F. Ramirez Rochac and N. Zhang, "Feature extraction in hyperspectral imaging using adaptive feature selection approach," *The Eighth International Conference on Advanced Computational Intelligence (ICACI2016)*, Chiang Mai, Thailand, pp. 36-40, 2016.

[7] J. Schmidhuber, “Deep learning in neural networks: an overview,” Neural Networks, vol. 61, pp. 85–117, 2015.

[8] H. R. Shahdoosti and F. Mirzapour, “Spectral–spatial feature extraction using orthogonal linear discriminant analysis for classification of hyperspectral data,” European Journal of Remote Sensing, vol. 50, no. 1, pp. 111-124, 2017.

[9] I. T. Jolliffe, “Principal Component Analysis,” Springer-Verlag, 2nd Edition, October, 2002.

[10] M. A. Carreria-Perpinan, “A review of dimension reduction techniques,” Technical Report CS-96-09, Dept. of Computer Science, University of Sheffield, January 1997.

[11] I. K. Fodor, “A survey of dimension reduction tehniques,”Technical Report UCRL-ID 148494, LLNL June 2002.

[12] E. J. Bjerrum. “SMILES enumeration as data augmentation for neural network modeling of molecules,” ArXiv e-prints, Mar. 2017.

[13] P. Cotae and S. Yalamanchili, “On the optimized sensor location performances in the presence of additive white Gaussian Noise,” *2008 IEEE International Conference on System of Systems Engineering*, Singapore, 2008.

[14] I. Goodfellow, Y. Bengio, A. Courville: Deep learning. MIT Press, 2016.

[15] F. Chollet, et al.: Keras. https://github.com/fchollet/keras, 2015.