Compressive Classification via Deep Learning using Single-pixel Measurements

Jorge Bacca[1], Nelson Diaz[2], and Henry Arguello[1]

Universidad Industrial de Santander
[1]Department of Computer Science
[2]Department of Electrical Engineering
Cra 27 calle 9 Ciudad Universitaria
Bucaramanga, Santander, 680002, Colombia
henarfu@uis.edu.co

Abstract

Single-pixel camera (SPC) captures encoded projections of the scene in a unique detector such that the number of compressive projections is lower than the size of the image. Traditionally, classification is not performed in the compressive domain because it is necessary to recover the underlying image before to classification. Based on the success of Deep learning (DL) in classification approaches, this paper proposes to classify images using compressive measurements of SPC. Furthermore, the proposed DL approach designs the binary sensing matrix in the SPC to improve the classification accuracy. In particular, a whole neural network is trained to learn the SPC sensing matrix, in the first layer, and extracts features from the single-pixel compressive measurements. The proposed approach overcomes two approaches of the state-of-the-art in terms of classification accuracy.

Introduction

Single-pixel camera (SPC) encodes the scene $\mathbf{x} \in \mathbb{R}^{MN}$ using $K \ll MN$ binary-coded aperture, obtaining $\mathbf{y} \in \mathbb{R}^{K}$ compressive projections of the scene, resulting in a hardware compression system [1]. The SPC can be represented as a linear system $\mathbf{y} = \mathbf{\Phi x}$, where each row of the sensing matrix $\mathbf{\Phi}$ corresponds to a rearrange of binary-coded aperture in a given snapshot. Traditionally, to classify images using compressive measurements of the SPC it is necessary to reconstruct the underlying scene solving a relative expensive optimization problem. However, there are approaches in compressive sensing that perform classification in the compressive domain, for instance, using Support Vector Machines (SVM) [2, 3] or sparse subspace clustering [4].

Recently, deep learning (DL) has been introduced as an approach to classify images, yielding high accuracy [5]. In particular, DL has been introduced to classify compressed measurements, extracting deep features from a re-projected image $\tilde{\mathbf{x}} = \mathbf{\Phi}^T \mathbf{y}$ where $\tilde{\mathbf{x}}$ has the image size, so any well known convolutional neural network can be used to classify image [6]. Indeed, [7] proposed an End-to-End network to train the sensing matrix $\mathbf{\Phi}$ and the re-projected matrix $\mathbf{\Phi}^T$ using two fully-connected layers in the first layers. However, This method requires a large number of parameters to optimize, and it does not take into account the physical constraints imposed by the real architectures as SPC. Therefore, this paper proposes a DL compressive classification approach for single-pixel compressive measurements. The procedure is summarized

in two parts of a whole neural network. The first part designs the binary sensing matrix, which represents the coded aperture in the SPC. The second part comprises the layers of a neural network that extracts deep features without re-projected measurements, which reduces training time [8]. This work is organized as follows. Section 1 describes the SPC acquisition model. Section 2 defines the coded aperture design and describes the proposed DL classification approach denominated binary-NoP-Net. Section 3 shows the results that compare the proposed approach against methods of the state-of-the-art such as Random+CNN [6], and End-to-End [7]. Conclusions are reported in section 4.

1 SPC Compressive acquisition model

The single-pixel camera (SPC) is a compressive imager that encodes the scene using a binary coded aperture which block/unblock some pixel in the image, then the encoded field is concentrated in a single sensor. The compressive measurement $\mathbf{y} \in \mathbb{R}^K$ can be expressed mathematically as

$$\mathbf{y} = \mathbf{\Phi}\mathbf{x} + \boldsymbol{\omega}, \tag{1}$$

where $\mathbf{\Phi} \in \{-1, 1\}^{K \times MN}$ denotes the binary sensing matrix, where each row represents the binary coded aperture, $\mathbf{x} \in \mathbb{R}^{MN}$ represents the underlying image and $\boldsymbol{\omega} \in \mathbb{R}^K$ is the additive noise. Usually, the reconstruction of the underlying scene is achieved assuming that the image is sparse in a given domain; therefore, it can be obtained by solving the following optimization problem

$$\hat{\mathbf{x}} = \mathbf{\Psi}(\underset{\boldsymbol{\alpha}}{\operatorname{argmin}} \|\mathbf{y} - \mathbf{\Phi}\mathbf{\Psi}\boldsymbol{\alpha}\|_2^2 + \tau\|\boldsymbol{\alpha}\|_1), \tag{2}$$

where the underlying image $\mathbf{x} = \mathbf{\Psi}\boldsymbol{\alpha}$ is represented as the linear combination between the representation basis $\mathbf{\Psi}$ and the sparse coefficients $\boldsymbol{\alpha}$. $\|.\|_2^2$, and $\|.\|_1$ denote the squared ℓ_2 norm, and the ℓ_1 norm, respectively. τ corresponds to a regularization parameter. It is essential to highlight that the distribution of the coded aperture determines the quality of the measurements [9], and a time-consuming iterative algorithm is required to solve (2).

2 Classification and coded aperture design using deep learning

The proposed approach introduces a supervised classifier based on deep learning (DL), that classifies images in the compressed domain, avoiding the reconstruction procedure of solving (2). Figure 1 depicts the proposed coupled neural network. The approach is divided into two stages. First, the training stage, where the coded apertures and the parameters of the classification network are learned. Second, the testing stage, where the compressed measurements are captured and classified with the learned parameters.

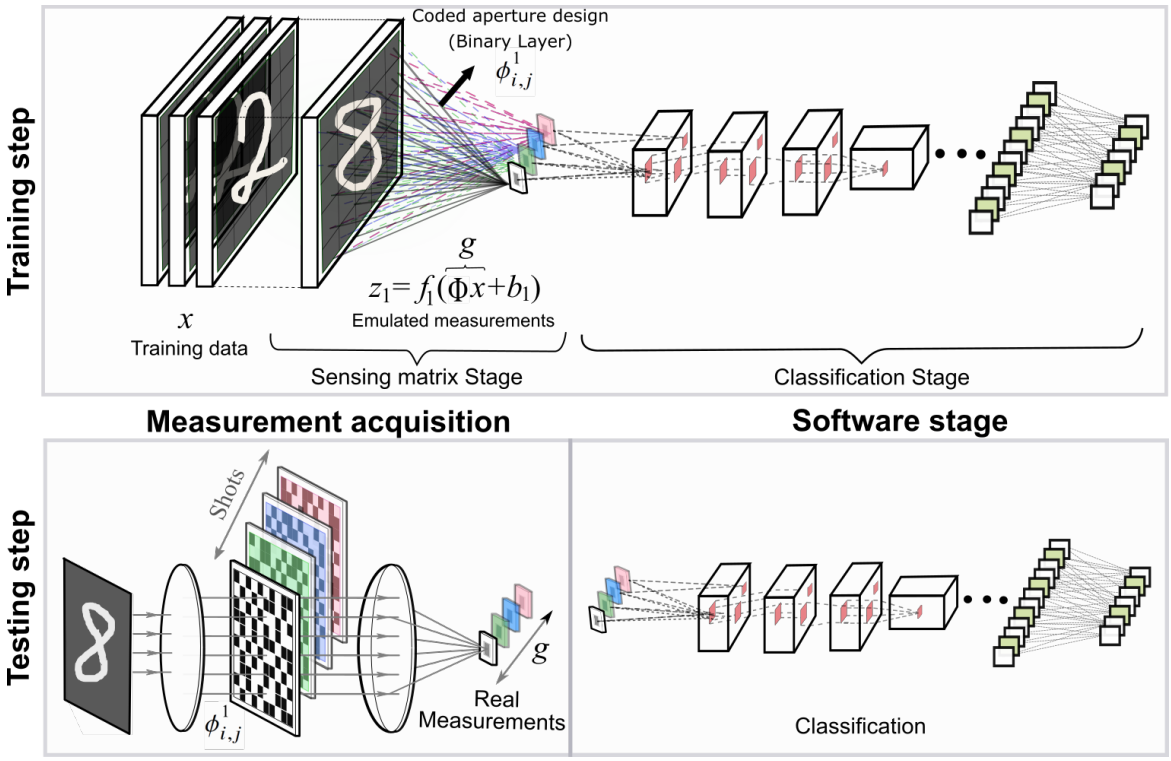

Figure 1: Training and testing stages for a coupled neural network, using single-pixel compressive measurements. The training is divided into two sub-stages, first learning the sensing matrix and second learn the parameter of the non-linear classifier. In the testing, the learned coded apertures are employed to obtain new compressive measurements. Those measurements are classified using the trained classifier network.

Training stage

The training stage is composed of the coupled neural network divided into two blocks. The first block is a binary fully connected layer which learns the implementable coded apertures; for this purpose, a penalization term E_{binary} is used to impose binary values. The second block is a non-linear classification network that learns the classification parameters of the network using a classification cost function E_{learning}. The coupled neural network is trained as a whole using a set of L test images $\{\mathbf{x}_\ell\}_{\ell=1}^{L}$, and the labels of the training images $\{\mathbf{d}_\ell\}_{\ell=1}^{L}$ to compute the sensing matrix $\mathbf{\Phi}$ and the parameters of the classifier $\boldsymbol{\theta}$.

Deep learning classification approach

The following corresponds to join the minimization cost function and learning problem penalized by binary sensing matrix regularization

$$\underset{\mathbf{\Phi},\boldsymbol{\theta}}{\operatorname{argmin}} \left\{ E_{\text{learning}} + \mu E_{\text{binary}} \right\}, \tag{3}$$

where μ denotes the regularization parameters that balance the trade-off between the joint learning problem, and penalization term.

Coded Aperture Design

The binary entries of the sensing matrix, which corresponds to the coded aperture, are learned introducing a penalty term, given by

$$E_{\text{binary}} = \sum_{k=1}^{K} \sum_{n=1}^{MN} (1 + \mathbf{\Phi}_{k,n})^2 (1 - \mathbf{\Phi}_{k,n})^2, \tag{4}$$

notice that the penalization term promotes a sensing matrix with values -1 or 1, which is implementable in a real experimental setup [10].

Classification Strategy

The second block of the training stage corresponds to the classification network. In detail, given the training data, the following loss function $\mathcal{L}\left(\mathcal{M}_{\boldsymbol{\theta}}(\mathbf{\Phi}\mathbf{x}_\ell), \mathbf{d}_\ell\right)$, that measure the error between the true label and the estimated by the inference operator where $\mathcal{M}_{\boldsymbol{\theta}}(.)$ corresponds to the non-linear classification network is given by

$$E_{\text{learning}} = \frac{1}{L} \sum_{\ell=1}^{L} \mathcal{L}(\mathcal{M}_{\boldsymbol{\theta}}(f_1(\mathbf{\Phi}\mathbf{x}_\ell + \mathbf{b}_1)), \mathbf{d}_\ell), \tag{5}$$

where $\mathcal{L}(\mathbf{z}_\ell, \mathbf{d}_\ell) = -\left[\mathbf{d}_\ell \log(\mathbf{z}_\ell) + (\mathbf{1} - \mathbf{d}_\ell) \log(\mathbf{1} - \mathbf{z}_\ell)\right]$ is the categorical cross entropy loss function, with $\mathbf{z}_\ell$ as the result of the classification operator for the ℓ-th image, i.e., $\mathbf{z}_\ell = \mathcal{M}_{\boldsymbol{\theta}}(f_1(\mathbf{\Phi}\mathbf{x}_\ell + \mathbf{b}_1))$. Therefore, the whole training method is summarized in the following loss function penalized by the term that promotes binary sensing matrix

$$\{\mathbf{\Phi}, \boldsymbol{\theta}\} = \underset{\mathbf{\Phi}, \boldsymbol{\theta}}{\operatorname{argmin}} \frac{1}{L} \sum_{\ell=1}^{L} \mathcal{L}(\mathcal{M}_{\boldsymbol{\theta}}(f_1(\mathbf{\Phi}\mathbf{x}_\ell + \mathbf{b}_1)), \mathbf{d}_\ell) + \mu \sum_{k=1}^{K} \sum_{n=1}^{MN} (1 + \mathbf{\Phi}_{k,n})^2 (1 - \mathbf{\Phi}_{k,n})^2. \tag{6}$$

With the structure of the classification operator, the eq. (6) can be solved using of the-state-of-art methods, including stochastic gradient descent (sgd) [11], min-batch gradient descent, gradient descent with Momentum [12], or Adam [13]. It is important to highlight that, after solving (6), the learned values of $\mathbf{\Phi}$ are designed specifically for the classification task and are binary values which can be used in the coded aperture of the SPC.

Testing stage

In the testing stage, there are two sub-stages. The first is a hardware sub-stage where each row of the sensing matrix $\mathbf{\Phi}$, learned in the previous training stage, are used as a coded aperture to acquire new compressive measurements $\mathbf{g} = \mathbf{\Phi}\mathbf{x}$. Then, the same learned network, starting from the second layer, can be used as a classification operator $\mathcal{M}_{\boldsymbol{\theta}}(.)$, whose output is given by

$$\mathbf{z} = \mathcal{M}_{\boldsymbol{\theta}}(f_1(\underbrace{\mathbf{g}}_{\mathbf{\Phi}\mathbf{x}} + \mathbf{b}_1)). \tag{7}$$

Configuration of the classification network

The proposed classification network $\mathcal{M}_\theta(\cdot)$ extracts deep features from the compressive measurements. Unlike [6, 7], the proposed network does not require a re-projected matrix to maintain the size of the original image. For that reason, the proposed method employs a smaller fully connected layer followed by an element-wise activation function like ReLU or a sigmoid [14], followed by convolutional layers, and max-pooling layers used to reduce dimensionality, finalizing with fully connected layers with soft-max function. The advantage of this approach is that it requires fewer parameters, which minimize overfitting risk [15] compared with [6, 7].

3 Results

This section analyzes the performance of the proposed coupled convolutional neural network called Binary-NoP-Net to classify images using SPC compressive measurements. It is compared against state-of-the-art DL classification models, including Random+CNN [6], and End-to-End [7]. In particular, Random+CNN uses random sensing matrix $\mathbf{\Phi}$ to obtain the compressive measurements and then uses a CNN to extract features from the re-projected measurements $\mathbf{\Phi}^T\mathbf{y}$. On the other hand, the End-to-End method learns a sensing matrix $\mathbf{\Phi}$ and also the re-projection matrix using two fully connected layers. It is worth noting that, End-to-End is not implementable in the SPC because the entries of the sensing matrix could converge to real values; however, it is included for comparison with the other approaches.

Simulation scenario

Two datasets are used in this paper to test the performance of the proposed method against the two approaches of the state-of-the-art. The first dataset is MNIST[1], and the second dataset is CIFAR-10[2] [16]. Both datasets have 60,000 images. Each dataset is divided into 10,000 training images, and 50,000 testing images. The MNIST dataset corresponds to images of digits from 0 to 9, with a size of 28×28 pixels. The CIFAR-10 dataset is composed of RGB images with a size of 32×32. The results for each database are averaged in 5 trial runs. In the testing step, the learned sensing matrix is utilized to acquire compressive measurements that were contaminated with additive noise corresponding to 30 dB signal-to-noise ratio (SNR). The three methods are trained with Adam algorithm [13]. The number of epochs is 100, and the learning rate is 0.001. In the proposed method, the hyper-parameter μ, which balances the trade-off between the joint learning problem, and the penalization term that promotes binary entries in the sensing matrix corresponds to 0.01. All parameters were obtained using cross-validation.
The simulations were computed using Matlab 2018a on an Intel Xeon E5-2697 2.6 GHz CPU with 192 GB RAM, coupled with an Nvidia Quadro K6000 12 GB GPU.

[1]Available at http://yann.lecun.com/exdb/mnist/
[2]Available at https://www.cs.toronto.edu/ kriz /cifar.html

MNIST database

Figure 2(a) depicts the configuration of the coupled neural network Binary-NoP-Net for MNIST dataset. The proposed method uses a binary layer succeed by three fully connected layers using ReLU as non-linear operator, and a 10-class softmax classifier, as is shown in Fig. 2(a). The Random+CNN and the End-to-End use a modification of LeNet-5 model [17], which is shown in Fig. 2(b). Notice that all network configurations have the same three final layers for this data set.

The average classification accuracy for the MNIST dataset is depicted in Table 1. The sensing ratios in this experiment correspond to $\gamma = \{0.01, 0.05, 0.01, 0.25\}$ or equivalently to number of snapshots $K = [8, 39, 78, 196]$. The results in boldface denote the best approach, and the underlined results represent the second-best result. The accuracy of the proposed approach increases as the sensing ratio increases. In particular, the proposed method obtains better result for low sensing ratios 1% or 5% compared with the implementable Random+CNN approach. One of the advantages of Binary-NoP-Net is the reduction in the number of parameters of the network to be trained. Specifically, Table 2 depicts the training time per epochs using the MNIST database and three methods. Notice that the proposed approach is faster in terms of computational time than the methods of the-state-of-the-art because the reduction in the number of layers, decrements the number of parameters.

Table 1: Average classification accuracy for different sensing ratios with the MNIST data set, and for the three methods.

Sensing Ratio (γ)	**Shots** (L)	**Data set**	**Methods**		
			Random+CNN	End-to-End	Binary-NoP-Net
0.25	196	Training	**99.99 %**	**99.99 %**	**99.99 %**
		Testing	98.32 ± 0.06 %	**98.48 ± 0.04 %**	97.37 ± 0.13 %
0.1	78	Training	99.95%	**99.99 %**	99.24 %
		Testing	97.01 ± 0.14 %	**98.29 ± 0.02 %**	95.88 ± 0.09%
0.05	39	Training	**99.99 %**	98.78%	98.97 %
		Testing	94.82 ± 0.09 %	**98.09 ± 0.04** %	94.95 ± 0.14 %
0.01	8	Training	89.98%	**97.19 %**	90.63 %
		Testing	58.94 ± 0.20 %	**95.18 ± 0.04** %	87.75 ± 0.35 %

Table 2: Training time in seconds per epoch with the MNIST data set, and for the three methods.

Sensing Ratio (γ)	**Shots** (L)	**Methods**		
		Random+CNN	End-to-End	Binary-NoP-Net
0.25	196	6.67	7.87	**4.33**
0.1	78	6.60	7.80	**4.29**
0.05	39	6.55	7.68	**4.25**
0.01	8	6.53	7.67	**4.20**

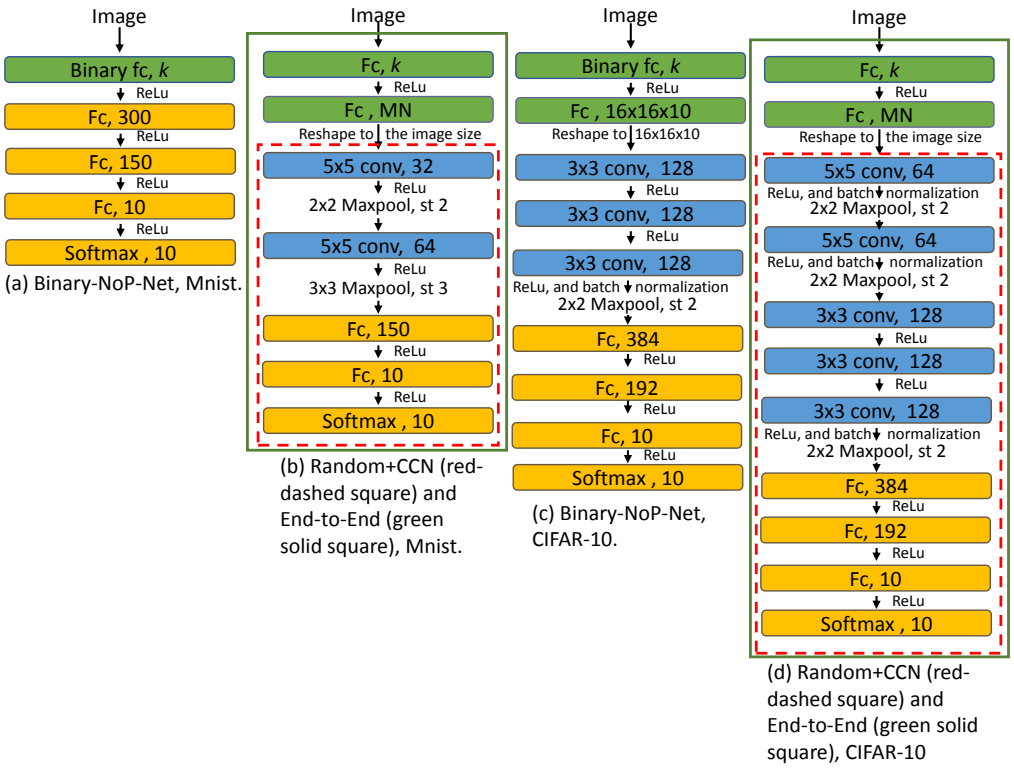

Figure 2: Layer description of the three neural network methods used with MNIST and CIFAR-10 datasets, (a) and (c) show the layer description of Binary-NoP-Net. (b) and (d) the layer configuration of Random+CNN and End-to-End. Inside the red-dashed rectangle is the layer configuration for Random+CNN whose input is $\hat{\mathbf{x}} = \mathbf{\Phi}^T\mathbf{y}$ (b) uses MNIST and (d) utilizes CIFAR-10. Inside the green-solid rectangle is the layer configuration for End-to-End (b) uses MNIST and (d) utilizes CIFAR-10.

CIFAR-10 database

Figure 2(c) shows the setup of the Binary-NoP-Net, which corresponds to the proposed coupled neural network for CIFAR-10 dataset. The classification block of Random+CNN and the End-to-End are a modification of AlexNet [5], which is depicted in Figure 2(d). Due to the ramification of the network and the dataset, the weights of layers 3 to 11 were initialized separately of the first two layers with learned weights from the training stage, using 50 epochs, and the same hyper-parameters used in MNIST dataset. The input of the binary-NoP-Net is the single-pixel compressive measurements. The method extracts the features using a small, fully connected layer, and the output of the last layer is rearranged into a 3D structure, followed by a convolutional layer, as is shown in Fig. 2(c). Notice that all network configurations have the same seven final layers.

The overall accuracy of the proposed Binary-NoP-Net is compared against methods of the-state-of-the-art Random+CNN and End-to-End in Table. 3. The same

sensing ratios $\gamma = [0.01, 0.05, 0.01, 0.25]$ of the preceding experiment were utilized with the CIFAR-10 dataset, which are equivalent to number of snapshots $K = [8, 39, 78, 196]$. The results in boldface denote the best approach, and the underlined results represent the second-best result. Notice that for the Binary-NoP-Net the higher the sensing ratios, the higher the classification accuracy. The proposed approach overcomes the implementable approach Ramdom+CNN in all the sensing ratio and the End-to-End for high sensing ration 0.1 and 0.25. Table 4 shows the training time in seconds using the database CIFAR-10. Notice that the proposed approach Binary-NoP-Net requires less computational time than the-state-of-the-art approaches.

Table 3: Average classification accuracy for different sensing ratios with the CIFAR-10 data set, and for the three methods.

Sensing	**Shots**	**Data**	**Methods**		
Ratio (γ)	(L)	**set**	Random+CNN	End-to-End	Binary-NoP-Net
0.25	768	Training	<u>99.82%</u>	**100 %**	98.24 %
		Testing	45.12 ± 0.24 %	<u>57.12 ± 0.40 %</u>	**63.45 ± 2.41 %**
0.1	307	Training	**92.55%**	<u>89.12</u> %	88.15 %
		Testing	40.84 ± 0.52 %	55.59 ± 0.55 %	**58.14 ± 1.11 %**
0.05	154	Training	<u>89.78%</u>	**91.89 %**	85.47 %
		Testing	34.25 ± 0.54 %	**54.34 ± 0.73 %**	<u>50.94 ± 0.98 %</u>
0.01	31	Training	<u>88.54%</u>	**90.64 %**	82.14 %
		Testing	30.47 ± 0.54 %	**50.89 ± 0.52** %	<u>40.59 ± 3.05 %</u>

Table 4: Training time in seconds per epoch with the CIFAR data set, and for the three methods.

Sensing	**Shots**	**Methods**		
Ratio (γ)	(L)	Random+CNN	End-to-End	Binary-NoP-Net
0.25	196	<u>158.5</u>	190.2	**82.6**
0.1	78	<u>157.5</u>	165.2	**80**
0.05	39	<u>156.8</u>	158.3	**78.9**
0.01	8	<u>156.3</u>	157.2	**78.2**

4 Conclusions

The coupled neural network Binary-NoP-Net was introduced, which learns concurrently the sensing matrix and the parameters of the non-linear classifier using the compressive measurements of the SPC. Then, the proposed approach obtains the compressive measurements of SPC using the learned sensing matrix, and the inference task is performed using the trained classification network. The proposed approach was compared with the-state-of-the-art methods Random+CNN, and End-to-End. Two datasets were used to evaluate the proposed approach MNIST and CIFAR-10. The proposed approach shows that when the sensing ratio increases, the classification increases using both datasets. The Binary-NoP-Net is overcome by the End-to-End

approach using the MNIST. However, the latter is not straightforward implementable into the SPC due to real valued entries of the sensing matrix. With the CIFAR-10 dataset, the proposed approach overcomes the-state-of-the-art approaches when the sensing ratio is increased. In terms of computational time, the proposed method is approximately twice faster than Random+CNN and End-to-End. Another advantage of the proposed method is that it requires less number of the parameter in the CNN model.

References

[1] M. F. Duarte, M. A. Davenport, D. Takhar, J. N. Laska, T. Sun, K. F. Kelly, and R. G. Baraniuk, "Single-pixel imaging via compressive sampling," *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 83–91, March 2008.

[2] J. M. Ramirez and H. Arguello, "Multiresolution compressive feature fusion for spectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, pp. 1–12, 2019.

[3] J. M. Ramirez and H. Arguello, "Spectral image classification from multi-sensor compressive measurements," *IEEE Transactions on Geoscience and Remote Sensing*, pp. 1–11, 2019.

[4] Carlos Hinojosa, Jorge Bacca, and Henry Arguello, "Coded aperture design for compressive spectral subspace clustering," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 6, pp. 1589–1600, 2018.

[5] 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.

[6] Suhas Lohit, Kuldeep Kulkarni, and Pavan Turaga, "Direct inference on compressive measurements using convolutional neural networks," in *2016 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2016, pp. 1913–1917.

[7] E Zisselman, A Adler, and M Elad, "Compressed learning for image classification: A deep neural network approach," *Processing, Analyzing and Learning of Images, Shapes, and Forms*, vol. 19, pp. 1, 2018.

[8] J. Bacca, C. Correa, E. Vargas, S. Castillo, and H. Arguello, "Compressive classification from single pixel measurements via deep learning," in *2019 IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, Oct. 2019 2019, pp. 1–6.

[9] Claudia V Correa, Henry Arguello, and Gonzalo R Arce, "Spatiotemporal blue noise coded aperture design for multi-shot compressive spectral imaging," *JOSA A*, vol. 33, no. 12, pp. 2312–2322, 2016.

[10] Jorge Luis Bacca-Quintero, Héctor Miguel Vargas-García, Daniel Ricardo Molina-Velasco, and Henry Arguello-Fuentes, "Single pixel compressive spectral polarization imaging using a movable micro-polarizer array," *Revista Facultad de Ingeniería Universidad de Antioquia*, , no. 88, pp. 91–99, 2018.

[11] Léon Bottou, "Large-scale machine learning with stochastic gradient descent," in *Proceedings of COMPSTAT'2010*, pp. 177–186. Springer, 2010.

[12] Ilya Sutskever, James Martens, George Dahl, and Geoffrey Hinton, "On the importance of initialization and momentum in deep learning," in *International conference on machine learning*, 2013, pp. 1139–1147.

[13] Diederik P Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[14] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals, "Understanding deep learning requires rethinking generalization," *CoRR*, vol. abs/1611.03530, 2016.

[15] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[16] Alex Krizhevsky and Geoffrey Hinton, "Learning multiple layers of features from tiny images," Tech. Rep., Citeseer, 2009.

[17] Yann LeCun, Léon Bottou, Yoshua Bengio, Patrick Haffner, et al., "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.