

Lensless Image Classification for Machine-based Decision Making

Ganghun Kim

Department of Electrical and Computer Engineering
University of Utah

Brian Rodriguez

Department of Electrical and Computer Engineering
University of Utah

Rajesh Menon

Department of Electrical and Computer Engineering
University of Utah

Abstract—Deep Learning (DL) has accelerated advancements in Image Classification via convolutional neural networks (CNNs). However, these image classification tasks have been widely trained on images taken with typical cameras; human-centric images. Here, we present a CNN trained using data taken by a single CMOS image sensor with no lens. We created a dataset of lensless images comprised of handwritten digits taken from the MNIST dataset. Then, we trained a CNN on this dataset and were able to show that for 10 digits, the CNN is able to classify lensless images with 96.6% accuracy.

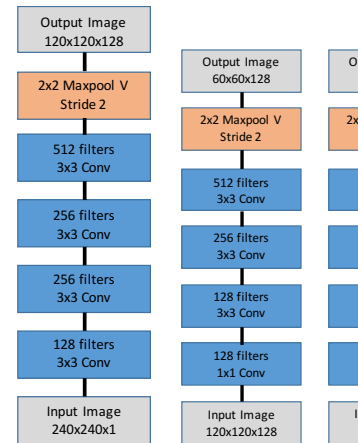
I. INTRODUCTION

Wide-scale deep learning algorithms have pushed image classification tasks to its limits. State of the art architectures have been able to classify human-centric images with astonishing accuracy, but human-centric images require expensive cameras [1], [2]. Recently, lensless imaging has been gaining traction [3]–[5]. These lensless images are taken using a low cost and light-weight image sensor without the use of a camera lens. Previously, we demonstrated that machine learning algorithms can accurately classify lensless images for 2 digits with 99% accuracy using support vector machines (SVMs) [6]. Here, we extend this idea and use DL to classify lensless images for 10 digits, accurately. Moreover, we trained a CNN on a dataset of lensless images that we captured using a lensless camera. Our lensless camera consists of a simple CMOS image sensor. We used a liquid-crystal display (LCD) that displayed images of hand-written digits taken from the open-sourced MNIST dataset [7]. The LCD rests 250mm away from the CMOS sensor with an exposure time of 150ms and averaged over 100 frames to reduce noise. Using this procedure, we captured 70,000 lensless images and created a dataset with appropriate labels. These lensless images are 640x480 (307,200 pixels per image) which for scalability purposes are resized to 240x240. Our results demonstrate that CNNs can accurately classify multiple classes of data taken directly from a lensless camera.

II. BACKGROUND

III. NETWORK ARCHITECTURE

We propose a CNN architecture that is able to classify lensless images at a 96.6% accuracy. Our architecture is comprised of two main sections; feature section and a classifying section. The feature section has 4 subsections containing convolutional layers and max pooling layers. Each convolutional layer is followed by batch normalization and a ReLU activation function [8], [9]. The 1x1 convolutional layers are a special case to this rule. These 1x1 layers are used for dimensionality reduction and do not utilize batch normalization or a ReLU activation function. Following the feature section is a classifying section. This section contains two fully-connected layers, where the last fully-connected layer classifies between the 10 different classes. We also employ dropout with a probability of .5 to prevent overfitting in the network [10].



IV. TRAINING METHODOLOGY

We created our network using Pytorch, a highly extensible deep learning framework [11]. Our network was trained using stochastic gradient descent with a momentum of .9 on a single NVidia Tesla V100 GPU

[12], [13]. We used a learning rate of .001 and a mini batch size of 16. The learning rate was decayed when the loss plateaued; decaying one time over 40 epochs. The input data is gray-scaled (by default), images are resized to 240x240 and randomly horizontally flipped. 5,400 images were taken from each class and split into two groups, training and testing groups; 3,200 training images and 2,200 testing images. Testing images are used for cross-validation; they serve to benchmark our model and test for overfitting in the network.

V. RESULTS

We tested a lot of different configurations for our final network. Our earlier experimental networks actually saw the highest accuracy than some of our later networks. Our later networks employed ideas such as Alpha Dropout and SELU activation functions [14]. We noticed that using extra learnable parameters actually hindered our models ability to learn. The highest accuracy we found using these new methods of learning was 82%. Our final network was able achieve a 96.6% accuracy and converged in fewer epochs. Meaning, the network was able to learn the necessary features quicker, in less time. While other gradient optimizers were tested such as Adam and Adagrad they did not converge accordingly [12], [15]. We found that in all of our experiments, stochastic gradient decent was the best optimizer and allowed our network to converge to its global minimum.

VI. CONCLUSION

Our results show promise for lensless imaging and a cost effective alternative for cameras in machine-based decision making.

REFERENCES

- [1] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, 2014.
- [2] C. Szegedy, S. Ioffe, and V. Vanhoucke, "Inception-v4, inception-resnet and the impact of residual connections on learning," *CoRR*, vol. abs/1602.07261, 2016.
- [3] M. S. Asif, A. Ayremlou, A. C. Sankaranarayanan, A. Veeraraghavan, and R. G. Baraniuk, "Flatcam: Thin, bare-sensor cameras using coded aperture and computation," *CoRR*, vol. abs/1509.00116, 2015.
- [4] P. R. Gill and D. G. Stork, "Lensless ultra-miniature imagers using odd-symmetry spiral phase gratings," in *Imaging and Applied Optics*, p. CW4C.3, Optical Society of America, 2013.
- [5] G. Kim, K. Isaacson, R. Palmer, and R. Menon, "Lensless photography with only an image sensor," *CoRR*, vol. abs/1702.06619, 2017.
- [6] G. Kim, S. Kapetanovic, R. Palmer, and R. Menon, "Lensless-camera based machine learning for image classification," *CoRR*, vol. abs/1709.00408, 2017.
- [7] Y. LeCun and C. Cortes, "MNIST handwritten digit database," 2010.
- [8] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *CoRR*, vol. abs/1502.03167, 2015.
- [9] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Proceedings of the 27th International Conference on International Conference on Machine Learning*, ICML'10, (USA), pp. 807–814, Omnipress, 2010.
- [10] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, pp. 1929–1958, 2014.
- [11] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," in *NIPS-W*, 2017.
- [12] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2014.
- [13] I. Sutskever, J. Martens, G. Dahl, and G. Hinton, "On the importance of initialization and momentum in deep learning," in *Proceedings of the 30th International Conference on International Conference on Machine Learning - Volume 28*, ICML'13, pp. III–1139–III–1147, JMLR.org, 2013.
- [14] G. Klambauer, T. Unterthiner, A. Mayr, and S. Hochreiter, "Self-normalizing neural networks," *CoRR*, vol. abs/1706.02515, 2017.
- [15] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," Tech. Rep. UCB/EECS-2010-24, EECS Department, University of California, Berkeley, Mar 2010.