# MNIST Classification with Deep Convolutional Neural Networks

**Aaron G. Alphonsus**
Department of Computer Science
South Dakota School of Mines and Technology
Rapid City, SD, 57701
aaron.alphonsus@mines.sdsmt.edu

## Abstract

Deep Neural Networks have regularly achieved state-of-the-art results especially in the field of image recognition. AlexNet in 2012, was one such network which achieved a near 50% reduction in the error rate of the previous state-of-the-art. In this project we have utilized some of the core architectural elements of AlexNet and used it to classify the MNIST dataset. Using TensorFlow, we were able to achieve a near state-of-the-art result of 99.2% accuracy. **NEED MORE?**

## 1 Introduction

The objective of this project was to pick a framework and dataset, and get a deep neural network trained and its performance tested. We took a few stabs at this with different frameworks and datasets and realized there was a steep learning curve to it. As this was our first time doing a machine learning project we decided to narrow the scope of the project classifying the introductory-level MNIST [1] dataset using the popular TensorFlow framework.

In Dr. McGough's lectures on convolutional networks [2], we were introduced to the history of the convolutional neural network and how they exploded on the scene. It was here that we saw the architectural features of AlexNet and learned how influential it was to the field at the time. The network from the original paper can be seen in figure 1. Note that the computation is divided into two pipelines showing the authors' use of 2 GPUs in training their network.

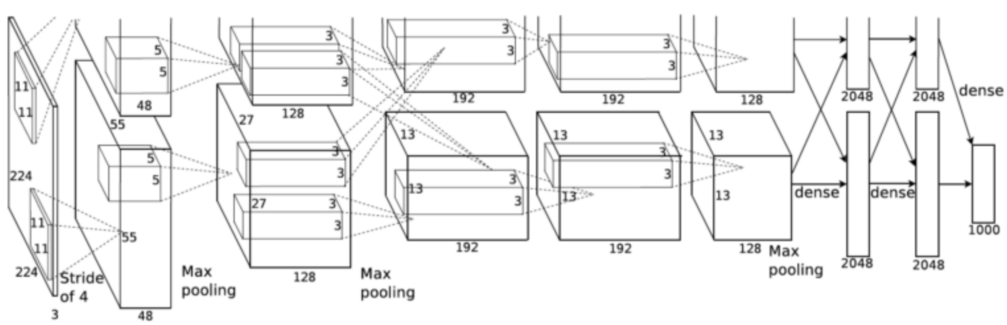


Figure 1: AlexNet neural network architecture

The paper by Krizhevsky proposed using a Deep Convolutional Neural Network for the task of image classification. The dataset that they used was the ImageNet dataset [3]. Looking at the results they obtained, we thought we would get some good results using it for a simple image dataset like MNIST.

31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA.

We began by implementing a simple single-layer neural network using TensorFlow to get used to the platform [4]. This yielded a modest 92% accuracy rate on the dataset. This is not great for the MNIST dataset, but creating the network did give us an end-to-end solution and it also got us familiar with the concepts and syntax.

We then moved on to implementing a deep convolutional network making use of dropout, ReLU neurons, successive convolution and pooling layers, and a densely connected final layer [5]. These were all features that were also used in AlexNet [6]. This model was a lot more successful, achieving an accuracy of 99.2%.

After we had the deep network implemented, we tested a few things out like different gradient descent optimizers, editing the network structure, and getting rid of dropout. We also wanted to work on being able to share our model without needing the user to spend time retraining it. Having the smaller model really helped here, as it meant we could test out edits without having to spend a lot of time waiting for the deep network to train. We weren't able to optimize the model further, but we were able to implement saving and restoring of the trained model.

## 2 Methods and Materials

Intro paragraph here providing an overview of the subsections

### 2.1 The Dataset

The MNIST dataset is popularly referred to as a "sanity check" dataset. This is because it is a relatively small dataset with not many categories. You can test out your new architecture quickly with it before turning it loose on your bigger datasets. This also means that it is a popular dataset among machine learning beginners. Having been unsuccessful getting a more complex dataset working, we decided to target MNIST in our project.

MNIST is a computer vision dataset. Each example is a gray-scale 28x28 pixel image and contains a handwritten digit from 0 to 9. Each digit has been centered in the 28x28 image. An example of the handwritten digits contained in the MNIST dataset can be seen in figure 2. The dataset has a training set of 60,000 examples and a test set of 10,000 examples. Since each image is a 0-9 digit, it is 10 category dataset.

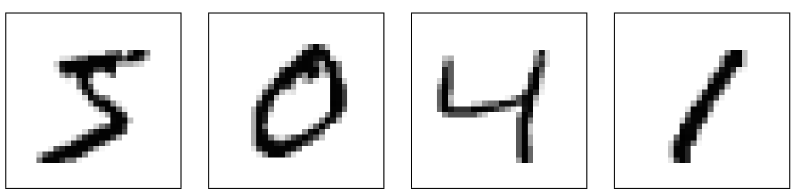

Figure 2: Examples of the handwritten digits in MNIST

### 2.2 Network Architecture

As mentioned earlier, our network architecture uses a lot of elements from AlexNet, as do a lot of neural networks that have come after it. AlexNet contains 60 million parameters and 650,000 neurons. It has 5 convolutional layers, 3 of which are followed by max-pooling, 3 fully connected layers, with a final 1000-way softmax. Looking at the features from AlexNet, we implemented a neural network with the following features in its network architecture:

- ReLU for the activation functions.
- 2 convolution layers, each followed by max pooling.
- A final densely-connected layer.
- Linear

The architecture for this network can be seen in figure **XYZ**

#### 2.2.1 Variables

#### 2.2.2 Convolution and Pooling

#### 2.2.3 Densely Connected Layer

### 2.3 Other Elements of the Model

The dropout regularization method to reduce over-fitting. Trying out different gradient descent alphas (Adam optimizer)

### 2.4 Saving and Restoring

## 3 Results

Using our deep convolutional network, we were able to achieve respectable results on the MNIST dataset. Comparing it to past results, as recorded the creator of the dataset Yann LeCun, our accuracy of 99.2% puts us among the best, albeit somewhere near the lower middle of the pack. As expected, our single-layer model isn't good enough for this dataset when compared to the state-of-the-art.

Convergence, cross entropy graphs for single and cnn

Comparison to published results

## 4 Discussion

In working on this project we used TensorFlow to train a deep network and were able to get respectable results on a popular dataset. We looked at the influential AlexNet architecture and implemented several features of its network architecture including dropout, ReLU neurons, successive convolution and pooling layers, and a densely connected layer at the end.

For future work on improving results with this model, we would like to look at some commonly-used data augmentation techniques such as random rotation, shear, and shift. This was a technique used by Krizhevsky as a method of increasing the robustness of their model.

Working on this project also served as a good introduction both in understanding the architecture of a deep convolutional network and getting familiar with the TensorFlow framework. We followed along with an introductory TensorFlow tutorial to get introduced to these new concepts as well as the low-level APIs.

The next step in continuing to develop our understanding of deep learning would be to try to incorporate some real-time visualization tools. Some exciting tools we have seen being used to visualize data during the training process are TensorBoard and matplotlib.

## References


[1] Y. LeCun, C. Cortes, and C. J.C. Burges, "The mnist database of handwritten digits." [Online]. Available: http://yann.lecun.com/exdb/mnist/

[2] L. Fei-Fei, J. Johnson, and S. Yeung, "Lecture 5: Convolutional neural networks." [Online]. Available: http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture5.pdf

[3] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2012, pp. 1097–1105. [Online]. Available: http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf

[4] G. B. Team, "Mnist for ml beginners." [Online]. Available: https://www.tensorflow.org/versions/r1.2/get_started/mnist/beginners

[5] ——, "Deep mnist for experts." [Online]. Available: https://www.tensorflow.org/versions/r1.2/get_started/mnist/pros

[6] G. Seif, "Deep learning for image recognition: why it's challenging, where we've been, and what's next." [Online]. Available: https://towardsdatascience.com/deep-learning-for-image-classification-why-its-challenging-where-we-ve-been-and-what-s-next-93b56948fcef

# 5 Submission of papers to NIPS 2017

NIPS requires electronic submissions. The electronic submission site is

`https://cmt.research.microsoft.com/NIPS2017/`

Please read carefully the instructions below and follow them faithfully.

## 5.1 Style

Papers to be submitted to NIPS 2017 must be prepared according to the instructions presented here. Papers may only be up to eight pages long, including figures. This does not include acknowledgments and cited references which are allowed on subsequent pages. Papers that exceed these limits will not be reviewed, or in any other way considered for presentation at the conference.

The margins in 2017 are the same as since 2007, which allow for $\sim 15\%$ more words in the paper compared to earlier years.

Authors are required to use the NIPS LaTeX style files obtainable at the NIPS website as indicated below. Please make sure you use the current files and not previous versions. Tweaking the style files may be grounds for rejection.

## 5.2 Retrieval of style files

The style files for NIPS and other conference information are available on the World Wide Web at

`http://www.nips.cc/`

The file `nips_2017.pdf` contains these instructions and illustrates the various formatting requirements your NIPS paper must satisfy.

The only supported style file for NIPS 2017 is `nips_2017.sty`, rewritten for LaTeX $2_\varepsilon$. **Previous style files for LaTeX 2.09, Microsoft Word, and RTF are no longer supported!**

The new LaTeX style file contains two optional arguments: `final`, which creates a camera-ready copy, and `nonatbib`, which will not load the `natbib` package for you in case of package clash.

At submission time, please omit the `final` option. This will anonymize your submission and add line numbers to aid review. Please do *not* refer to these line numbers in your paper as they will be removed during generation of camera-ready copies.

The file `nips_2017.tex` may be used as a "shell" for writing your paper. All you have to do is replace the author, title, abstract, and text of the paper with your own.

The formatting instructions contained in these style files are summarized in Sections 6, 7, and 8 below.

# 6 General formatting instructions

The text must be confined within a rectangle 5.5 inches (33 picas) wide and 9 inches (54 picas) long. The left margin is 1.5 inch (9 picas). Use 10 point type with a vertical spacing (leading) of 11 points. Times New Roman is the preferred typeface throughout, and will be selected for you by default. Paragraphs are separated by ½ line space (5.5 points), with no indentation.

The paper title should be 17 point, initial caps/lower case, bold, centered between two horizontal rules. The top rule should be 4 points thick and the bottom rule should be 1 point thick. Allow ¼ inch space above and below the title to rules. All pages should start at 1 inch (6 picas) from the top of the page.

For the final version, authors' names are set in boldface, and each name is centered above the corresponding address. The lead author's name is to be listed first (left-most), and the co-authors' names (if different address) are set to follow. If there is only one co-author, list both author and co-author side by side.

Please pay special attention to the instructions in Section 8 regarding figures, tables, acknowledgments, and references.

# 7 Headings: first level

All headings should be lower case (except for first word and proper nouns), flush left, and bold.

First-level headings should be in 12-point type.

## 7.1 Headings: second level

Second-level headings should be in 10-point type.

### 7.1.1 Headings: third level

Third-level headings should be in 10-point type.

**Paragraphs** There is also a `\paragraph` command available, which sets the heading in bold, flush left, and inline with the text, with the heading followed by 1 em of space.

# 8 Citations, figures, tables, references

These instructions apply to everyone.

## 8.1 Citations within the text

The `natbib` package will be loaded for you by default. Citations may be author/year or numeric, as long as you maintain internal consistency. As to the format of the references themselves, any style is acceptable as long as it is used consistently.

The documentation for `natbib` may be found at

`http://mirrors.ctan.org/macros/latex/contrib/natbib/natnotes.pdf`

Of note is the command `\citet`, which produces citations appropriate for use in inline text. For example,

```
\citet{hasselmo} investigated\dots
```

produces

Hasselmo, et al. (1995) investigated…

If you wish to load the `natbib` package with options, you may add the following before loading the `nips_2017` package:

```
\PassOptionsToPackage{options}{natbib}
```

If `natbib` clashes with another package you load, you can add the optional argument `nonatbib` when loading the style file:

```
\usepackage[nonatbib]{nips_2017}
```

As submission is double blind, refer to your own published work in the third person. That is, use "In the previous work of Jones et al. [4]," not "In our previous work [4]." If you cite your other papers that are not widely available (e.g., a journal paper under review), use anonymous author names in the citation, e.g., an author of the form "A. Anonymous."

Table 1: Sample table title

| Part | | |
| --- | --- | --- |
| Name | Description | Size ($\mu$m) |
| Dendrite | Input terminal | $\sim$100 |
| Axon | Output terminal | $\sim$10 |
| Soma | Cell body | up to $10^6$ |

### 8.2 Footnotes

Footnotes should be used sparingly. If you do require a footnote, indicate footnotes with a number[1] in the text. Place the footnotes at the bottom of the page on which they appear. Precede the footnote with a horizontal rule of 2 inches (12 picas).

Note that footnotes are properly typeset *after* punctuation marks.[2]

### 8.3 Figures

All artwork must be neat, clean, and legible. Lines should be dark enough for purposes of reproduction. The figure number and caption always appear after the figure. Place one line space before the figure caption and one line space after the figure. The figure caption should be lower case (except for first word and proper nouns); figures are numbered consecutively.

You may use color figures. However, it is best for the figure captions and the paper body to be legible if the paper is printed in either black/white or in color.

Figure 3: Sample figure caption.

### 8.4 Tables

All tables must be centered, neat, clean and legible. The table number and title always appear before the table. See Table 1.

Place one line space before the table title, one line space after the table title, and one line space after the table. The table title must be lower case (except for first word and proper nouns); tables are numbered consecutively.

Note that publication-quality tables *do not contain vertical rules.* We strongly suggest the use of the `booktabs` package, which allows for typesetting high-quality, professional tables:

`https://www.ctan.org/pkg/booktabs`

This package was used to typeset Table 1.

[1]Sample of the first footnote.
[2]As in this example.

# 9 Final instructions

Do not change any aspects of the formatting parameters in the style files. In particular, do not modify the width or length of the rectangle the text should fit into, and do not change font sizes (except perhaps in the **References** section; see below). Please note that pages should be numbered.

# 10 Preparing PDF files

Please prepare submission files with paper size "US Letter," and not, for example, "A4."

Fonts were the main cause of problems in the past years. Your PDF file must only contain Type 1 or Embedded TrueType fonts. Here are a few instructions to achieve this.

- You should directly generate PDF files using `pdflatex`.
- You can check which fonts a PDF files uses. In Acrobat Reader, select the menu Files>Document Properties>Fonts and select Show All Fonts. You can also use the program `pdffonts` which comes with `xpdf` and is available out-of-the-box on most Linux machines.
- The IEEE has recommendations for generating PDF files whose fonts are also acceptable for NIPS. Please see `http://www.emfield.org/icuwb2010/downloads/IEEE-PDF-SpecV32.pdf`
- `xfig` "patterned" shapes are implemented with bitmap fonts. Use "solid" shapes instead.
- The `\bbold` package almost always uses bitmap fonts. You should use the equivalent AMS Fonts:

  ```
  \usepackage{amsfonts}
  ```

  followed by, e.g., `\mathbb{R}`, `\mathbb{N}`, or `\mathbb{C}` for $\mathbb{R}$, $\mathbb{N}$ or $\mathbb{C}$. You can also use the following workaround for reals, natural and complex:

  ```
  \newcommand{\RR}{I\!\!R} %real numbers
  \newcommand{\Nat}{I\!\!N} %natural numbers
  \newcommand{\CC}{I\!\!\!\!C} %complex numbers
  ```

  Note that `amsfonts` is automatically loaded by the `amssymb` package.

If your file contains type 3 fonts or non embedded TrueType fonts, we will ask you to fix it.

## 10.1 Margins in LaTeX

Most of the margin problems come from figures positioned by hand using `\special` or other commands. We suggest using the command `\includegraphics` from the `graphicx` package. Always specify the figure width as a multiple of the line width as in the example below:

```
\usepackage[pdftex]{graphicx} ...
\includegraphics[width=0.8\linewidth]{myfile.pdf}
```

See Section 4.4 in the graphics bundle documentation (`http://mirrors.ctan.org/macros/latex/required/graphics/grfguide.pdf`)

A number of width problems arise when LaTeX cannot properly hyphenate a line. Please give LaTeX hyphenation hints using the `\-` command when necessary.

### Acknowledgments

Use unnumbered third level headings for the acknowledgments. All acknowledgments go at the end of the paper. Do not include acknowledgments in the anonymized submission, only in the final paper.

# References

References follow the acknowledgments. Use unnumbered first-level heading for the references. Any choice of citation style is acceptable as long as you are consistent. It is permissible to reduce the font size to `small` (9 point) when listing the references. **Remember that you can go over 8 pages as long as the subsequent ones contain *only* cited references.**


[1] Alexander, J.A. & Mozer, M.C. (1995) Template-based algorithms for connectionist rule extraction. In G. Tesauro, D.S. Touretzky and T.K. Leen (eds.), *Advances in Neural Information Processing Systems 7*, pp. 609–616. Cambridge, MA: MIT Press.

[2] Bower, J.M. & Beeman, D. (1995) *The Book of GENESIS: Exploring Realistic Neural Models with the GEneral NEural SImulation System.* New York: TELOS/Springer–Verlag.

[3] Hasselmo, M.E., Schnell, E. & Barkai, E. (1995) Dynamics of learning and recall at excitatory recurrent synapses and cholinergic modulation in rat hippocampal region CA3. *Journal of Neuroscience* **15**(7):5249-5262.