10-701 Final Project

Michael Lee
Department of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213
ml5@andrew.cmu.edu

Abstract

The abstract paragraph should be indented ½ inch (3 picas) on both the left- and right-hand margins. Use 10 point type, with a vertical spacing (leading) of 11 points. The word **Abstract** must be centered, bold, and in point size 12. Two line spaces precede the abstract. The abstract must be limited to one paragraph.

1 Left/Right Footprint Detection

(TODO - describe the task if there's time and space)

1.1 Image Preprocessing

1.1.1 Motivation

The 5000 training images for left and right footprints were always bounded by a rectangle of Gaussian noise and came in a variety of orientations and crops. The flexibility and expressivity of neural networks as a series of many nonlinear functions can both be its greatest strength but also its greatest weakness. While the object in this problem is to learn what left and right footprints look like respectively, a neural network may actually end up learning other additional things if trained on the raw data. For example, it might notice that left feet are more prone to be vertically oriented while right feet are more prone to be horizontally oriented, which is an artifact of our particular dataset. Therefore, all the training images and testing images were preprocessed in the following manner.

1.2 Method

All images were affected by Gaussian noise, first a mask of the foot print track (area within the parallel line boundary between the footprint and surrounding noise and white background) was created. This was done by blurring the image with two median filters (of size 5 x 5 then 10 x 10) and finding all the pixels below the pixel brightness of 255 (which extracted the white background). A median filter was used to maintain as much of the sharp edge between the footprint track and the white background as possible. Using a simple averaging Gaussian filter would have allowed outliers to affect the result of the filter more strongly (if most of the pixels in the filter bounding box were white background but a few were black, the result of the filter be pulled to below 255 where a median filter would generate a 255 result and include it outside of the footprint track region of the mask (TODO - reword)).

Then the edge of the mask were located using a Sobel filter, which combines the result from horizontal and vertical derivative operations which give rise to horizontal and vertical edges respectively. The longest edge was then found using MATLAB's function for finding connected components in an image, which always corresponded to the edges of the track what paralleled the sides of the footprint. The image was then rotated using the angle between that edge and the y axis of the image to get it in a vertical position. (While this method for orienting the images wasn't perfect and it left the ambiguity of the foot pointing upward or downward, the neural network was able to determine the determine the difference between the left and right feet reasonably well even still).

The final steps of the preprocessing step involved cropping the iamge to only include the footprint track (without the background) and resizing the image to feed into the neural network. Though the average imension of the training images was found to be 470 x 230, the final preprocessed image size was reduced to 235 x 115 due to GPU memory constraints.

(TODO - show series of preprocessing steps)

1.3 Learning algorithm

1.3.1 Library used

Keras with Tensorflow was selected due to its simple front-end interface and very powerful backend for building and training neural networks. Though it turns out that it is possible to achieve very decent accuracy with training a kernalized SVM with a RBF kernel with PCA features (95%), I made the personal choice to go with a neural network to gain hands on experience with deep learning through this project.

1.4 Network Architecture

For the simple left/right classification, the original MNIST architecture was trained from scratch on the training data. The rationale for starting off with this architecture was that recent, more popular but complex networks for Imagenet like Alexnet designed to do 1000 way classification would be must too large and be very likely to overfit.

In the first few attempts of training, it was found that the training and validation accuracies were stuck at 69% and 50% respectively regardless of the various changes made, such as trying Alexnet architecture, adding additional layers, etc. It turned out that although the Adagrad weight update method is often preferred over Adadelta due to its greater robustness to intial learning rate and often faster convergence (albeit at a poorer local minima), Adagrad performs very poorly on this dataset. As many others have found, Adagrad does better than Adagrad in a saddle point (TODO - ref) but Adadelta performs better at an error manifold that resembles the Beale's function (the large initial gradient makes Adagrad almost go unstable) and in the long valley (Adadelta breaks the symmetry and descends much faster). Though the figures only represent canonical 2D error manifolds, it still gives some insight as to how the various weight optimizers behave. The error manifold was probably ridden with very many local minima that the more conservative Adagrad approach could not escape out of.

(TODO - put in images from imgur and cite Alec Radford)

With a more suitable weight optimizer, the final architecture was taken to as follows in Figure (TODO - cite). It took the original MNIST architecture and added two back to back convolution layers after the max pooling layer, which is a common paradigm used in many Imagenet architectures like Alexnet and VGG.

(TODO - put in image of final network)

2 Footprint Matching

(TODO - describe the task if there's time and space)

2.1 Image Preprocessing

Image preprocessing of the training and testing images were necessary due to significant disparty between the two. While the original training images were always correctly oriented and the footprints very clear with no noise, the testing images were peppered with Gaussian noise and oriented in an arbitrary orientation. And while the testing images could either be of the left or right foot, the training images were always a right footprint.

Using ImageDataGenerator class of Python's Keras library, each training image was resized into a standard size for training with neural network. The same class was also used to create 20 additional images of a random orientation between 0 and 360 degrees with a random horizontal flip to allow them to be left or right foot. The total training size increased from 1,000 images to 20,000 images.

The Gaussian noise from the test images were removed and the remaining foot print was made into a binary image to mimic the clean images training images.

(TODO - insert sample training and testing images here)

2.2 Architecture and Methods

As Figure (TODO - cite) from Eugenio Culurciello demonstrates, the Inception v3 architecture was used as it seemed to be a good compromise between accuracy and parameters. Three different training methods were tried.

(TODO - insert diagram)

2.2.1 Training from Scratch

The first method attempted trained the Inception architecture from scratch, under the rationale that the Imagenet weights would have learned to differentiate color in the RGB Imagenet images. The training data was split into a 70/30 train/validation split, which was carefully done so that the training and validation distributions were the same. This meant that 70% of the augmented images for each of the 1000 classes were used for training and 30% of the augmented images for validation, as opposed to a random 70/30 split of the entire 20,000 image dataset. However, after training for 12 epochs, the many of the 10,000 test predictions consisted of only a subset of the training images (for example, training image 812 would repeated by be predicted eight times in a row).

This led to the hypothesis that the network was not expressive or deep enough.

2.2.2 Additional Layers

The next method involved inserting a couple dense layers at the top of the network to increase the network's ability to learn. The training and validation results improved (achieving 95% and 92%) respectively, but it did poorly on test data, only achieving 0.1% which corresponds to a random guess. So now it looked like the network was overfitting.

This was the test data was proprocessed to remove Gaussian noise and have binary footprint images, which brough up the test accuracy to 0.2%. In order to try and prevent overfitting, transfer learning was tried next.

2.2.3 Transfer Learning

The original Inception v3 architecture was used but now loaded with pretrained Imagenet weights. The first layer was changed however to accomodate grayscale images. The input images were resized to be 224 x 224 to mimic the original Imagenet images.

This new network was trained for 6 epochs and achieved extremely good training and validation accuracies but failed to achieve a high test accuracy, indicating overfitting once again.

2.3 Considerations

In the end, the highest achieved test accuracy was 0.2% with an Inception V3 base architecture trained from scratch with additional dense layers placed at the end.

Some additional techniques that would've been worthwhile to try include

- further preprocessing the images to get rid of some the solid black lines and rulers at the periphery of the image.
- orienting the test images vertically so it's not left to chance that one of the 20 augmentations would be at the exact orientation as the one arbitrarily oriented test image

3 Submission of papers to NIPS 2016

There is a new style file for papers submitted in 2016!

NIPS requires electronic submissions. The electronic submission site is

`https://cmt.research.microsoft.com/NIPS2016/`

Please read carefully the instructions below and follow them faithfully.

3.1 Style

Papers to be submitted to NIPS 2016 must be prepared according to the instructions presented here. Papers may only be up to eight pages long, including figures. Since 2009 an additional ninth page *containing only acknowledgments and/or cited references* is allowed. Papers that exceed nine pages will not be reviewed, or in any other way considered for presentation at the conference.

The margins in 2016 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.

3.2 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_2016.pdf` contains these instructions and illustrates the various formatting requirements your NIPS paper must satisfy.

The only supported style file for NIPS 2016 is `nips_2016.sty`, rewritten for LaTeX 2_{ε}. **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_2016.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 4, 5, and 6 below.

4 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 6 regarding figures, tables, acknowledgments, and references.

5 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.

5.1 Headings: second level

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

5.1.1 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.

6 Citations, figures, tables, references

These instructions apply to everyone.

6.1 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_2016` 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_2016}
```

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 (μm)
Dendrite	Input terminal	$\sim$100
Axon	Output terminal	$\sim$10
Soma	Cell body	up to 10^6

6.2 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]

6.3 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.

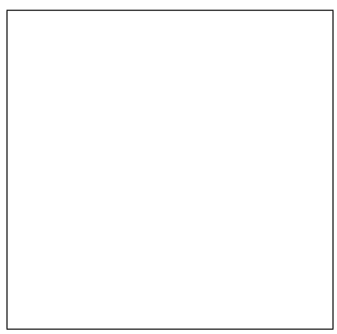

Figure 1: Sample figure caption.

6.4 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.

7 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.

8 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.

8.1 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 use a ninth page as long as it contains *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.