

# CNNS

## ... FOR SPEECH

[martin @ reddragon.ai](mailto:martin@reddragon.ai)  
[sam @ reddragon.ai](mailto:sam@reddragon.ai)

4 October 2017

WiFi : SG-Guest

Problems with Installation? **ASK!**

# PLAN OF ACTION

TODAY

- CNNs in pictures
- CNNs on speech data
- Launch into a CNN mini-challenge
- Get projects into Topics

# PLAN OF ACTION

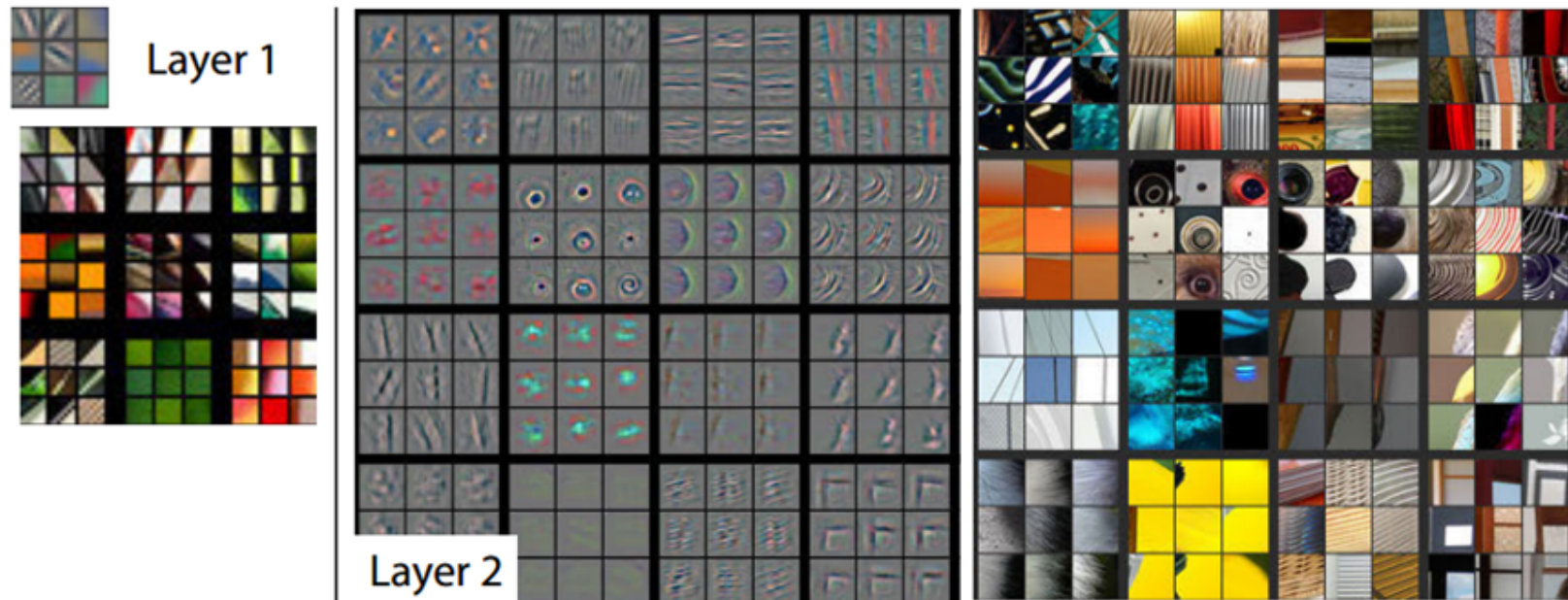
MONDAY

- Start with mini-challenge wrap-up
- CNN transfer learning
- Other CNN trickery...

# CNN REPRESENTATIONS

- ( Should have shown this before )
- Each successive layer ...
  - ... seems to learn 'higher level' representations
- All created by BackProp (no hand-crafted features)
- Surprisingly like the *actual brain*

# EARLY LEVELS

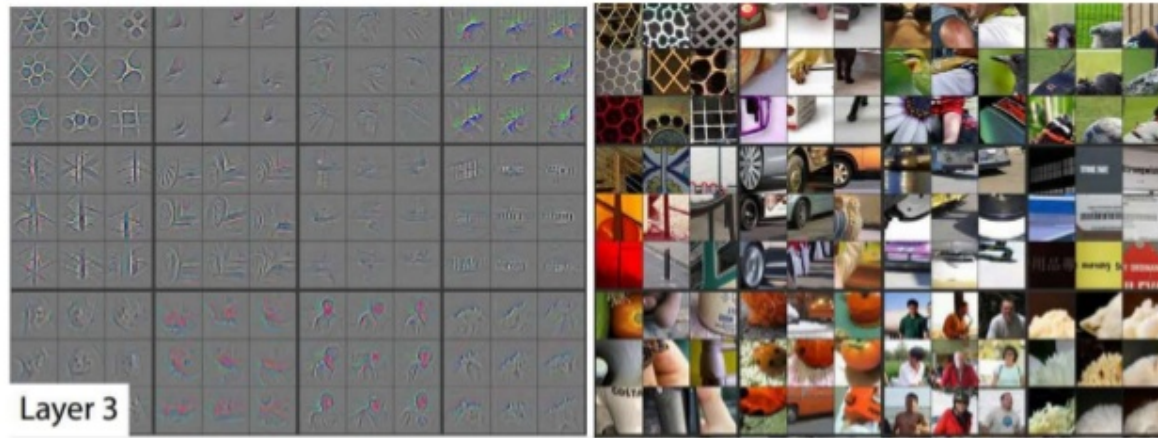


Visualizations of Layer 1 and 2. Each layer illustrates 2 pictures, one which shows the filters themselves and one that shows what part of the image are most strongly activated by the given filter. For example, in the space labeled Layer 2, we have representations of the 16 different filters (on the left)

Visualizing and Understanding Convolutional Networks (2013) (large download)

# MIDDLE LEVEL

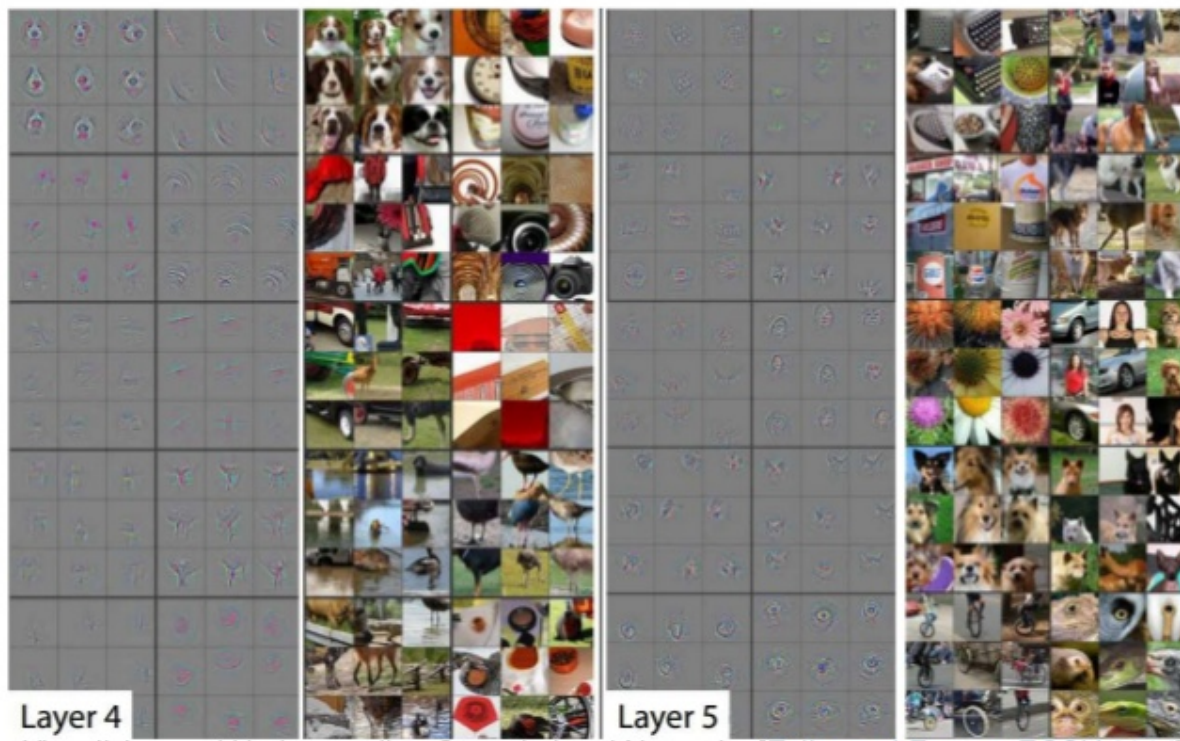
Layer 3





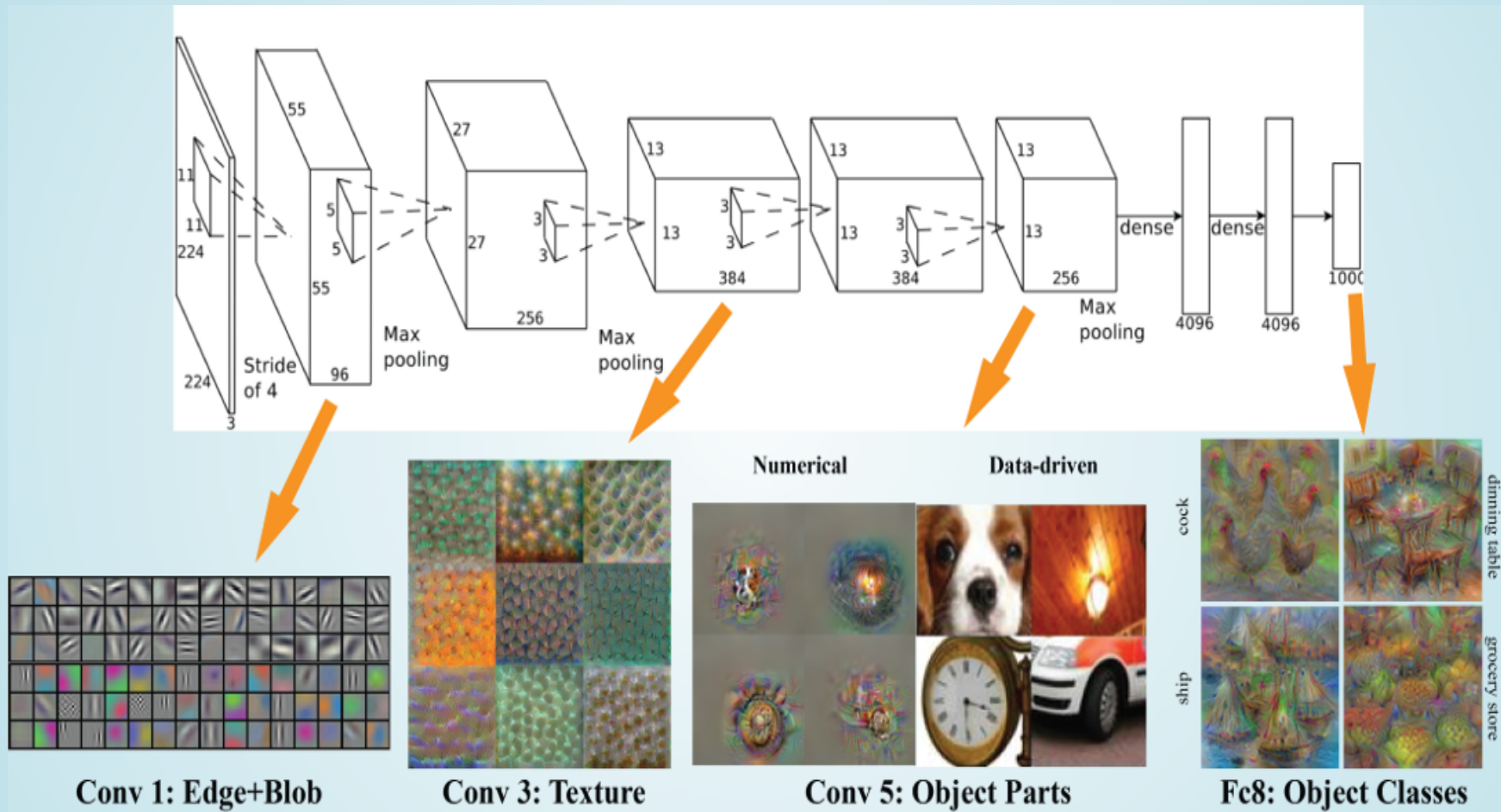
# LATER LEVELS

## Layer 4 and 5



Visualizing and Understanding Convolutional Networks [Zeiler and Fergus, ECCV 2014]

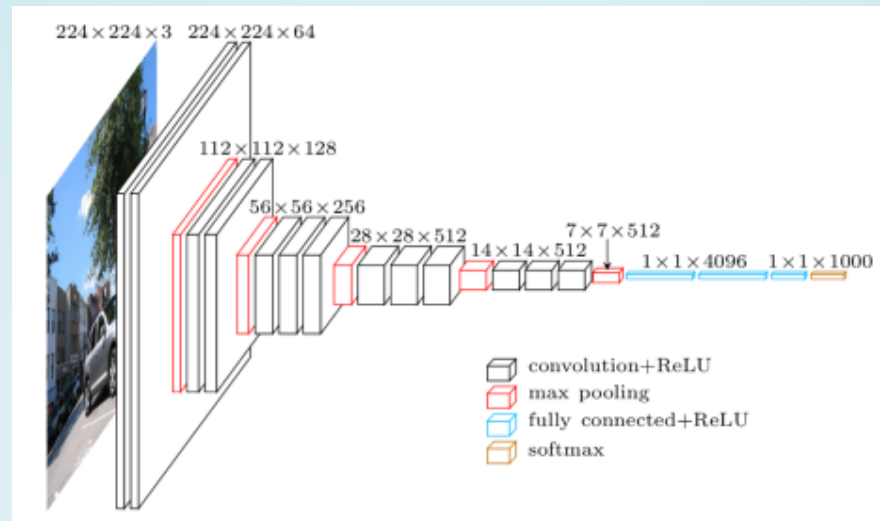
# ALEXNET



ImageNet Classification with Deep Convolutional Neural Networks (2012) (Alex Krizhevsky)



# VGG GENERIC

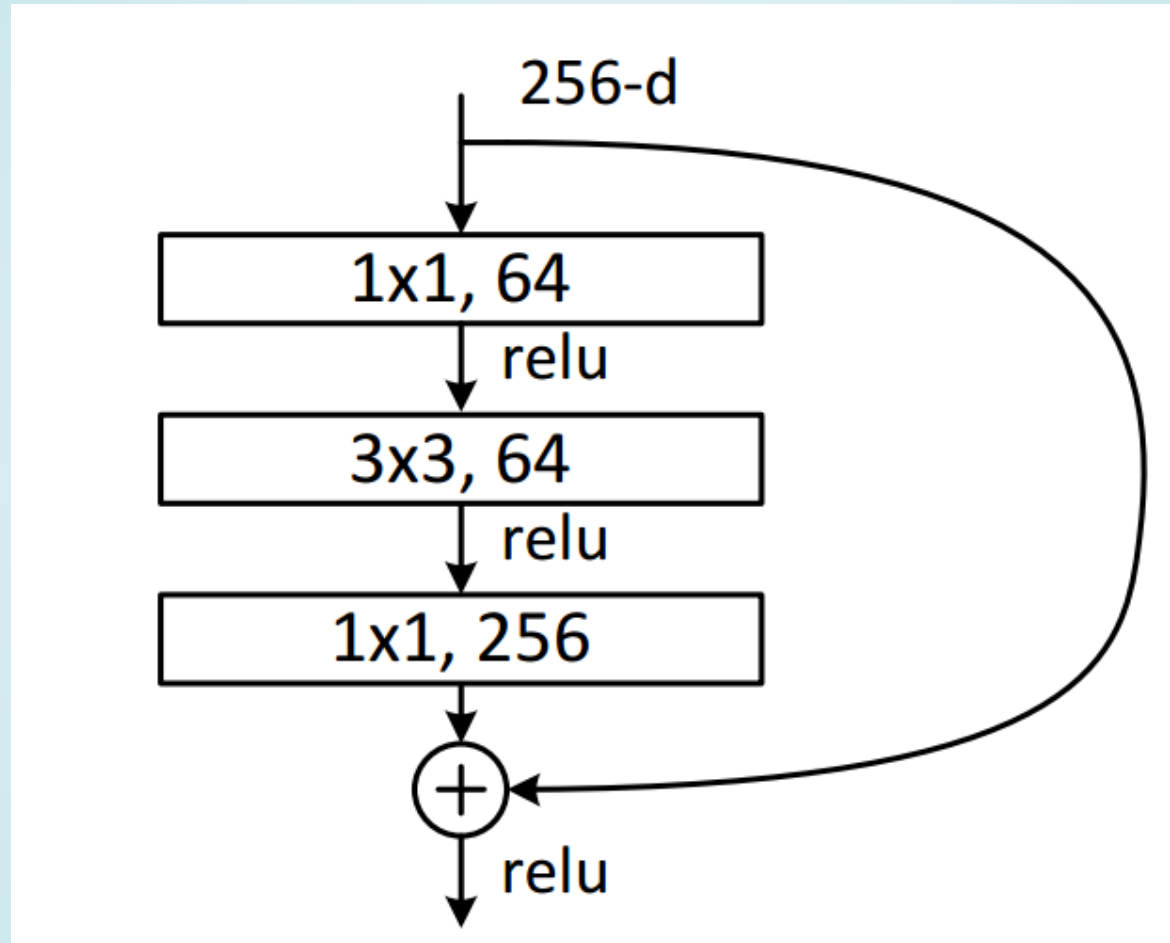


Very Deep Convolutional Networks for Large-Scale Image Recognition (2014)

# VGG TABLE

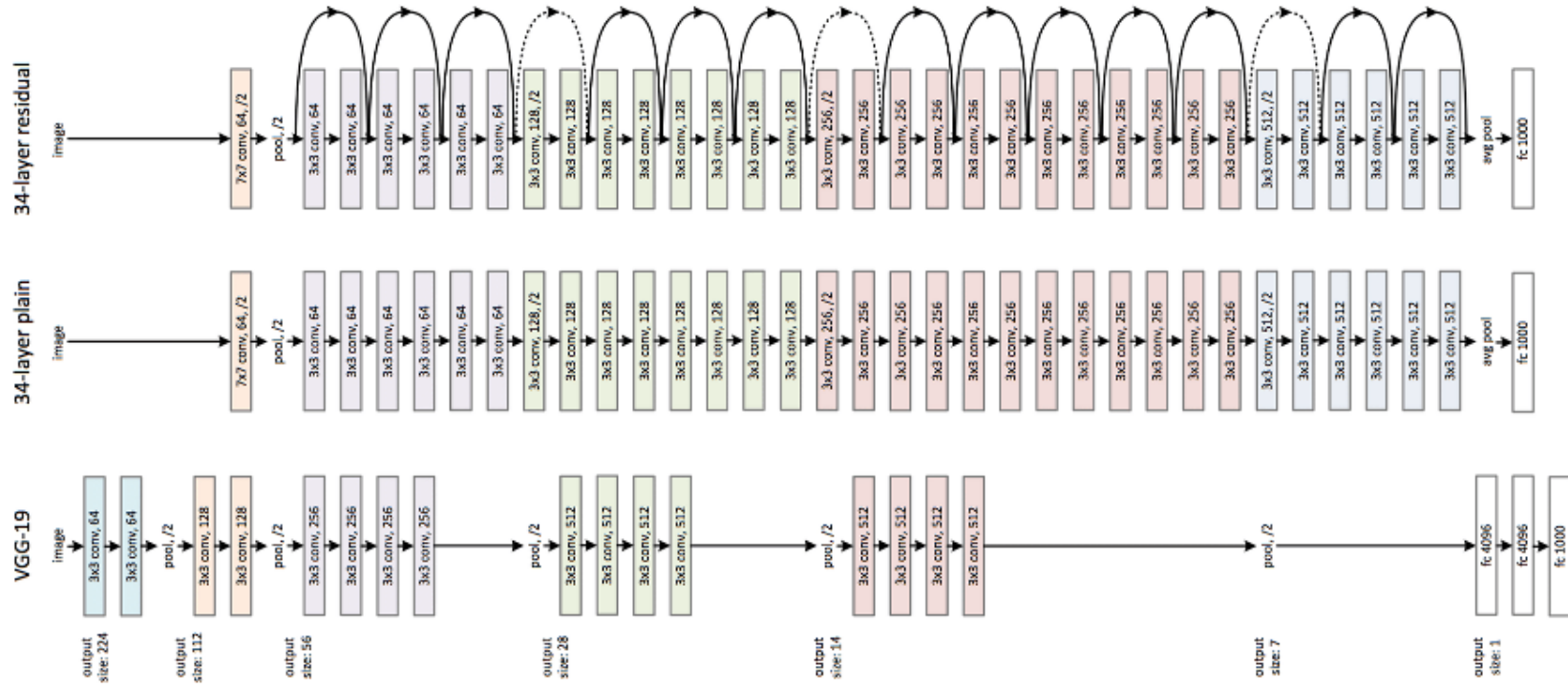
ConvNet Configuration					
A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
input ( $224 \times 224$ RGB image)					
conv3-64	conv3-64 LRN	conv3-64 <b>conv3-64</b>	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	conv3-128	conv3-128 <b>conv3-128</b>	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maxpool					
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 <b>conv1-256</b>	conv3-256 conv3-256 <b>conv3-256</b>	conv3-256 conv3-256 conv3-256 <b>conv3-256</b>
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 <b>conv1-512</b>	conv3-512 conv3-512 <b>conv3-512</b>	conv3-512 conv3-512 conv3-512 <b>conv3-512</b>
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 <b>conv1-512</b>	conv3-512 conv3-512 <b>conv3-512</b>	conv3-512 conv3-512 conv3-512 <b>conv3-512</b>
maxpool					
FC-4096					
FC-4096					
FC-1000					
soft-max					

# RESNET BLOCK



Deep Residual Learning for Image Recognition (2015) (Microsoft)

# RESNET VS VGG



# MORE CNNS ?

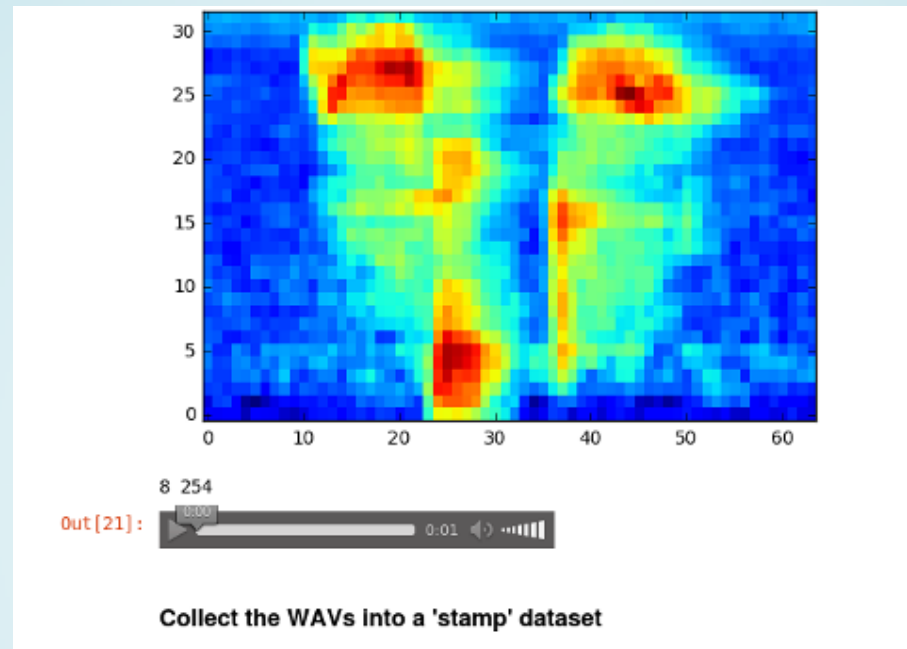
- Since CNNs are good at images ...
- ... make everything into images

# LET'S ABUSE A CNN

- For example : **Speech Recognition**
- Make this into an Image Recognition task
- <https://github.com/mdda/cnn-speech-mnist>



# SPEECH DATA 'STAMPS'



'cnn-speech-mnist/SpeechRecognition\_Data.ipynb'

# CNN SPEECH RECOGNITION

```
stamps.shape: (31, 64, 32)
```

```
labels.shape: (31,)
```

```
batch_input_fn sizing : (31, 64, 32, 1)
```

```
INFO:tensorflow:Starting evaluation at 2017-03-18-04:31:04
```

```
INFO:tensorflow:Evaluation [1/1]
```

```
INFO:tensorflow:Finished evaluation at 2017-03-18-04:31:05
```

```
{'accuracy': 1.0, 'loss': 0.0068151536, 'global_step': 3830}
```

```
'cnn-speech-mnist/SpeechRecognition_Learn.ipynb'
```

# - QUESTIONS -

MARTIN.ANDREWS @  
REDDRAGON.AI

My blog : <http://mdda.net/>

GitHub : [mdda](#)