

pretrained

January 2, 2019

1 Installation details

Mind that if you want to use your GPU (Nvidia CUDA compatible is needed), you need to install the GPU version of the package along with CUDA, Nvidia's package for interfacing between the user and the GPU, and cuDNN, a Deep Neural Network base package.

- CUDA: <http://www.nvidia.es/object/cuda-parallel-computing-es.html>
- cuDNN: <https://developer.nvidia.com/cudnn>

Notice that training NN with CPU is slower than using GPU, even though Keras/Tensorflow does some kind of parallelization of the operations using BLAS libraries (Basic Linear Algebra Subprograms). Therefore, GPU usage is advised for this session. But don't worry, the code for CPU and GPU is exactly the same except for one line where we define the execution.

2 Loading the model

```
In [1]: library(keras)
library(imager)
```

```
Loading required package: magrittr
```

```
Attaching package: 'imager'
```

```
The following object is masked from 'package:magrittr':
```

```
add
```

```
The following objects are masked from 'package:stats':
```

```
convolve, spectrum
```

```
The following object is masked from 'package:graphics':
```

```
frame
```

```
The following object is masked from 'package:base':
```

```
save.image
```

Loading Batch-Normalized Inception network

Reference:

- Batch normalization: Accelerating deep network training by reducing internal covariate shift(Loffe et al., 2015).

Based on Inceptionv3 network: rethinking the inception architecture for computer vision (Szegedy et al., 2015) <https://arxiv.org/abs/1512.00567>

```
In [4]: model <- application_inception_v3(include_top = TRUE, weights = "imagenet",
                                         input_tensor = NULL, input_shape = c(299,299,3), pooling = NULL,
                                         classes = 1000)
```

Now we check the model:

```
In [5]: model
```

Model

Layer (type)	Output Shape	Param #	Connected to
<hr/>			
input_2 (InputLayer)	(None, 299, 299, 0)		
conv2d_95 (Conv2D)	(None, 149, 149, 864)	input_2[0] [0]	
batch_normalization_95 (Batch Normalization)	(None, 149, 149, 96)	conv2d_95[0] [0]	
activation_95 (Activation)	(None, 149, 149, 0)	batch_normalization_95[0] [0]	
conv2d_96 (Conv2D)	(None, 147, 147, 9216)	activation_95[0] [0]	
batch_normalization_96 (Batch Normalization)	(None, 147, 147, 96)	conv2d_96[0] [0]	
activation_96 (Activation)	(None, 147, 147, 0)	batch_normalization_96[0] [0]	
conv2d_97 (Conv2D)	(None, 147, 147, 18432)	activation_96[0] [0]	
batch_normalization_97 (Batch Normalization)	(None, 147, 147, 192)	conv2d_97[0] [0]	
activation_97 (Activation)	(None, 147, 147, 0)	batch_normalization_97[0] [0]	
max_pooling2d_5 (MaxPooling2D)	(None, 73, 73, 64)	activation_97[0] [0]	
conv2d_98 (Conv2D)	(None, 73, 73, 5120)	max_pooling2d_5[0] [0]	
batch_normalization_98 (Batch Normalization)	(None, 73, 73, 240)	conv2d_98[0] [0]	
<hr/>			

```
activation_98 (Activation (None, 73, 73, 80 0           batch_normalization_98[0] [0]
-----
conv2d_99 (Conv2D)          (None, 71, 71, 19 138240 activation_98[0] [0]
-----
batch_normalization_99 (B (None, 71, 71, 19 576       conv2d_99[0] [0]
-----
activation_99 (Activation (None, 71, 71, 19 0           batch_normalization_99[0] [0]
-----
max_pooling2d_6 (MaxPooli (None, 35, 35, 19 0         activation_99[0] [0]
-----
conv2d_103 (Conv2D)          (None, 35, 35, 64 12288 max_pooling2d_6[0] [0]
-----
batch_normalization_103 ( (None, 35, 35, 64 192       conv2d_103[0] [0]
-----
activation_103 (Activatio (None, 35, 35, 64 0           batch_normalization_103[0] [
-----
conv2d_101 (Conv2D)          (None, 35, 35, 48 9216 max_pooling2d_6[0] [0]
-----
conv2d_104 (Conv2D)          (None, 35, 35, 96 55296 activation_103[0] [0]
-----
batch_normalization_101 ( (None, 35, 35, 48 144       conv2d_101[0] [0]
-----
batch_normalization_104 ( (None, 35, 35, 96 288       conv2d_104[0] [0]
-----
activation_101 (Activatio (None, 35, 35, 48 0           batch_normalization_101[0] [
-----
activation_104 (Activatio (None, 35, 35, 96 0           batch_normalization_104[0] [
-----
average_pooling2d_10 (Ave (None, 35, 35, 19 0         max_pooling2d_6[0] [0]
-----
conv2d_100 (Conv2D)          (None, 35, 35, 64 12288 max_pooling2d_6[0] [0]
-----
conv2d_102 (Conv2D)          (None, 35, 35, 64 76800 activation_101[0] [0]
-----
conv2d_105 (Conv2D)          (None, 35, 35, 96 82944 activation_104[0] [0]
-----
conv2d_106 (Conv2D)          (None, 35, 35, 32 6144 average_pooling2d_10[0] [0]
-----
batch_normalization_100 ( (None, 35, 35, 64 192       conv2d_100[0] [0]
-----
batch_normalization_102 ( (None, 35, 35, 64 192       conv2d_102[0] [0]
-----
batch_normalization_105 ( (None, 35, 35, 96 288       conv2d_105[0] [0]
-----
batch_normalization_106 ( (None, 35, 35, 32 96       conv2d_106[0] [0]
-----
activation_100 (Activatio (None, 35, 35, 64 0           batch_normalization_100[0] [
-----
```

```
activation_102 (Activation) (None, 35, 35, 64 0           batch_normalization_102[0] [  
-----  
activation_105 (Activation) (None, 35, 35, 96 0           batch_normalization_105[0] [  
-----  
activation_106 (Activation) (None, 35, 35, 32 0           batch_normalization_106[0] [  
-----  
mixed0 (Concatenate)      (None, 35, 35, 25 0           activation_100[0] [0]  
                           activation_102[0] [0]  
                           activation_105[0] [0]  
                           activation_106[0] [0]  
-----  
conv2d_110 (Conv2D)       (None, 35, 35, 64 16384     mixed0[0] [0]  
-----  
batch_normalization_110 ( (None, 35, 35, 64 192         conv2d_110[0] [0]  
-----  
activation_110 (Activation) (None, 35, 35, 64 0           batch_normalization_110[0] [  
-----  
conv2d_108 (Conv2D)       (None, 35, 35, 48 12288     mixed0[0] [0]  
-----  
conv2d_111 (Conv2D)       (None, 35, 35, 96 55296     activation_110[0] [0]  
-----  
batch_normalization_108 ( (None, 35, 35, 48 144         conv2d_108[0] [0]  
-----  
batch_normalization_111 ( (None, 35, 35, 96 288         conv2d_111[0] [0]  
-----  
activation_108 (Activation) (None, 35, 35, 48 0           batch_normalization_108[0] [  
-----  
activation_111 (Activation) (None, 35, 35, 96 0           batch_normalization_111[0] [  
-----  
average_pooling2d_11 (Ave (None, 35, 35, 25 0           mixed0[0] [0]  
-----  
conv2d_107 (Conv2D)       (None, 35, 35, 64 16384     mixed0[0] [0]  
-----  
conv2d_109 (Conv2D)       (None, 35, 35, 64 76800     activation_108[0] [0]  
-----  
conv2d_112 (Conv2D)       (None, 35, 35, 96 82944     activation_111[0] [0]  
-----  
conv2d_113 (Conv2D)       (None, 35, 35, 64 16384     average_pooling2d_11[0] [0]  
-----  
batch_normalization_107 ( (None, 35, 35, 64 192         conv2d_107[0] [0]  
-----  
batch_normalization_109 ( (None, 35, 35, 64 192         conv2d_109[0] [0]  
-----  
batch_normalization_112 ( (None, 35, 35, 96 288         conv2d_112[0] [0]  
-----  
batch_normalization_113 ( (None, 35, 35, 64 192         conv2d_113[0] [0]  
-----  
activation_107 (Activation) (None, 35, 35, 64 0           batch_normalization_107[0] [  
-----
```

```
-----  
activation_109 (Activation) (None, 35, 35, 64 0           batch_normalization_109[0] [  
-----  
activation_112 (Activation) (None, 35, 35, 96 0           batch_normalization_112[0] [  
-----  
activation_113 (Activation) (None, 35, 35, 64 0           batch_normalization_113[0] [  
-----  
mixed1 (Concatenate)      (None, 35, 35, 28 0           activation_107[0] [0]  
                           activation_109[0] [0]  
                           activation_112[0] [0]  
                           activation_113[0] [0]  
-----  
conv2d_117 (Conv2D)       (None, 35, 35, 64 18432     mixed1[0] [0]  
-----  
batch_normalization_117 ( (None, 35, 35, 64 192        conv2d_117[0] [0]  
-----  
activation_117 (Activation) (None, 35, 35, 64 0           batch_normalization_117[0] [  
-----  
conv2d_115 (Conv2D)       (None, 35, 35, 48 13824     mixed1[0] [0]  
-----  
conv2d_118 (Conv2D)       (None, 35, 35, 96 55296     activation_117[0] [0]  
-----  
batch_normalization_115 ( (None, 35, 35, 48 144        conv2d_115[0] [0]  
-----  
batch_normalization_118 ( (None, 35, 35, 96 288        conv2d_118[0] [0]  
-----  
activation_115 (Activation) (None, 35, 35, 48 0           batch_normalization_115[0] [  
-----  
activation_118 (Activation) (None, 35, 35, 96 0           batch_normalization_118[0] [  
-----  
average_pooling2d_12 (Ave (None, 35, 35, 28 0           mixed1[0] [0]  
-----  
conv2d_114 (Conv2D)       (None, 35, 35, 64 18432     mixed1[0] [0]  
-----  
conv2d_116 (Conv2D)       (None, 35, 35, 64 76800     activation_115[0] [0]  
-----  
conv2d_119 (Conv2D)       (None, 35, 35, 96 82944     activation_118[0] [0]  
-----  
conv2d_120 (Conv2D)       (None, 35, 35, 64 18432     average_pooling2d_12[0] [0]  
-----  
batch_normalization_114 ( (None, 35, 35, 64 192        conv2d_114[0] [0]  
-----  
batch_normalization_116 ( (None, 35, 35, 64 192        conv2d_116[0] [0]  
-----  
batch_normalization_119 ( (None, 35, 35, 96 288        conv2d_119[0] [0]  
-----  
batch_normalization_120 ( (None, 35, 35, 64 192        conv2d_120[0] [0]  
-----
```

```

activation_114 (Activation) (None, 35, 35, 64 0           batch_normalization_114[0] [
-----
activation_116 (Activation) (None, 35, 35, 64 0           batch_normalization_116[0] [
-----
activation_119 (Activation) (None, 35, 35, 96 0           batch_normalization_119[0] [
-----
activation_120 (Activation) (None, 35, 35, 64 0           batch_normalization_120[0] [
-----
mixed2 (Concatenate)      (None, 35, 35, 28 0           activation_114[0][0]
                                         activation_116[0][0]
                                         activation_119[0][0]
                                         activation_120[0][0]
-----
conv2d_122 (Conv2D)       (None, 35, 35, 64 18432     mixed2[0][0]
-----
batch_normalization_122 ( (None, 35, 35, 64 192         conv2d_122[0][0]
-----
activation_122 (Activation) (None, 35, 35, 64 0           batch_normalization_122[0] [
-----
conv2d_123 (Conv2D)       (None, 35, 35, 96 55296     activation_122[0][0]
-----
batch_normalization_123 ( (None, 35, 35, 96 288         conv2d_123[0][0]
-----
activation_123 (Activation) (None, 35, 35, 96 0           batch_normalization_123[0] [
-----
conv2d_121 (Conv2D)       (None, 17, 17, 38 995328    mixed2[0][0]
-----
conv2d_124 (Conv2D)       (None, 17, 17, 96 82944     activation_123[0][0]
-----
batch_normalization_121 ( (None, 17, 17, 38 1152         conv2d_121[0][0]
-----
batch_normalization_124 ( (None, 17, 17, 96 288         conv2d_124[0][0]
-----
activation_121 (Activation) (None, 17, 17, 38 0           batch_normalization_121[0] [
-----
activation_124 (Activation) (None, 17, 17, 96 0           batch_normalization_124[0] [
-----
max_pooling2d_7 (MaxPooling) (None, 17, 17, 28 0        mixed2[0][0]
-----
mixed3 (Concatenate)      (None, 17, 17, 76 0           activation_121[0][0]
                                         activation_124[0][0]
                                         max_pooling2d_7[0][0]
-----
conv2d_129 (Conv2D)       (None, 17, 17, 12 98304     mixed3[0][0]
-----
batch_normalization_129 ( (None, 17, 17, 12 384         conv2d_129[0][0]
-----
activation_129 (Activation) (None, 17, 17, 12 0           batch_normalization_129[0] [

```

```
-----  
conv2d_130 (Conv2D)      (None, 17, 17, 12 114688 activation_129[0] [0]  
-----  
batch_normalization_130 ( (None, 17, 17, 12 384 conv2d_130[0] [0]  
-----  
activation_130 (Activatio (None, 17, 17, 12 0 batch_normalization_130[0] [  
-----  
conv2d_126 (Conv2D)      (None, 17, 17, 12 98304 mixed3[0] [0]  
-----  
conv2d_131 (Conv2D)      (None, 17, 17, 12 114688 activation_130[0] [0]  
-----  
batch_normalization_126 ( (None, 17, 17, 12 384 conv2d_126[0] [0]  
-----  
batch_normalization_131 ( (None, 17, 17, 12 384 conv2d_131[0] [0]  
-----  
activation_126 (Activatio (None, 17, 17, 12 0 batch_normalization_126[0] [  
-----  
activation_131 (Activatio (None, 17, 17, 12 0 batch_normalization_131[0] [  
-----  
conv2d_127 (Conv2D)      (None, 17, 17, 12 114688 activation_126[0] [0]  
-----  
conv2d_132 (Conv2D)      (None, 17, 17, 12 114688 activation_131[0] [0]  
-----  
batch_normalization_127 ( (None, 17, 17, 12 384 conv2d_127[0] [0]  
-----  
batch_normalization_132 ( (None, 17, 17, 12 384 conv2d_132[0] [0]  
-----  
activation_127 (Activatio (None, 17, 17, 12 0 batch_normalization_127[0] [  
-----  
activation_132 (Activatio (None, 17, 17, 12 0 batch_normalization_132[0] [  
-----  
average_pooling2d_13 (Ave (None, 17, 17, 76 0 mixed3[0] [0]  
-----  
conv2d_125 (Conv2D)      (None, 17, 17, 19 147456 mixed3[0] [0]  
-----  
conv2d_128 (Conv2D)      (None, 17, 17, 19 172032 activation_127[0] [0]  
-----  
conv2d_133 (Conv2D)      (None, 17, 17, 19 172032 activation_132[0] [0]  
-----  
conv2d_134 (Conv2D)      (None, 17, 17, 19 147456 average_pooling2d_13[0] [0]  
-----  
batch_normalization_125 ( (None, 17, 17, 19 576 conv2d_125[0] [0]  
-----  
batch_normalization_128 ( (None, 17, 17, 19 576 conv2d_128[0] [0]  
-----  
batch_normalization_133 ( (None, 17, 17, 19 576 conv2d_133[0] [0]  
-----  
batch_normalization_134 ( (None, 17, 17, 19 576 conv2d_134[0] [0]
```

```
-----  
activation_125 (Activation) (None, 17, 17, 19 0      batch_normalization_125[0] [  
-----  
activation_128 (Activation) (None, 17, 17, 19 0      batch_normalization_128[0] [  
-----  
activation_133 (Activation) (None, 17, 17, 19 0      batch_normalization_133[0] [  
-----  
activation_134 (Activation) (None, 17, 17, 19 0      batch_normalization_134[0] [  
-----  
mixed4 (Concatenate)      (None, 17, 17, 76 0      activation_125[0] [0]  
                           activation_128[0] [0]  
                           activation_133[0] [0]  
                           activation_134[0] [0]  
-----  
conv2d_139 (Conv2D)       (None, 17, 17, 16 122880  mixed4[0] [0]  
-----  
batch_normalization_139 ( (None, 17, 17, 16 480      conv2d_139[0] [0]  
-----  
activation_139 (Activation) (None, 17, 17, 16 0      batch_normalization_139[0] [  
-----  
conv2d_140 (Conv2D)       (None, 17, 17, 16 179200  activation_139[0] [0]  
-----  
batch_normalization_140 ( (None, 17, 17, 16 480      conv2d_140[0] [0]  
-----  
activation_140 (Activation) (None, 17, 17, 16 0      batch_normalization_140[0] [  
-----  
conv2d_136 (Conv2D)       (None, 17, 17, 16 122880  mixed4[0] [0]  
-----  
conv2d_141 (Conv2D)       (None, 17, 17, 16 179200  activation_140[0] [0]  
-----  
batch_normalization_136 ( (None, 17, 17, 16 480      conv2d_136[0] [0]  
-----  
batch_normalization_141 ( (None, 17, 17, 16 480      conv2d_141[0] [0]  
-----  
activation_136 (Activation) (None, 17, 17, 16 0      batch_normalization_136[0] [  
-----  
activation_141 (Activation) (None, 17, 17, 16 0      batch_normalization_141[0] [  
-----  
conv2d_137 (Conv2D)       (None, 17, 17, 16 179200  activation_136[0] [0]  
-----  
conv2d_142 (Conv2D)       (None, 17, 17, 16 179200  activation_141[0] [0]  
-----  
batch_normalization_137 ( (None, 17, 17, 16 480      conv2d_137[0] [0]  
-----  
batch_normalization_142 ( (None, 17, 17, 16 480      conv2d_142[0] [0]  
-----  
activation_137 (Activation) (None, 17, 17, 16 0      batch_normalization_137[0] [  
-----
```

```
activation_142 (Activation) (None, 17, 17, 16 0           batch_normalization_142[0] [  
-----  
average_pooling2d_14 (Ave) (None, 17, 17, 76 0           mixed4[0] [0]  
-----  
conv2d_135 (Conv2D)      (None, 17, 17, 19 147456       mixed4[0] [0]  
-----  
conv2d_138 (Conv2D)      (None, 17, 17, 19 215040       activation_137[0] [0]  
-----  
conv2d_143 (Conv2D)      (None, 17, 17, 19 215040       activation_142[0] [0]  
-----  
conv2d_144 (Conv2D)      (None, 17, 17, 19 147456       average_pooling2d_14[0] [0]  
-----  
batch_normalization_135 (Batch Normalization) (None, 17, 17, 19 576 conv2d_135[0] [0]  
-----  
batch_normalization_138 (Batch Normalization) (None, 17, 17, 19 576 conv2d_138[0] [0]  
-----  
batch_normalization_143 (Batch Normalization) (None, 17, 17, 19 576 conv2d_143[0] [0]  
-----  
batch_normalization_144 (Batch Normalization) (None, 17, 17, 19 576 conv2d_144[0] [0]  
-----  
activation_135 (Activation) (None, 17, 17, 19 0           batch_normalization_135[0] [  
-----  
activation_138 (Activation) (None, 17, 17, 19 0           batch_normalization_138[0] [  
-----  
activation_143 (Activation) (None, 17, 17, 19 0           batch_normalization_143[0] [  
-----  
activation_144 (Activation) (None, 17, 17, 19 0           batch_normalization_144[0] [  
-----  
mixed5 (Concatenate)     (None, 17, 17, 76 0           activation_135[0] [0]  
                           activation_138[0] [0]  
                           activation_143[0] [0]  
                           activation_144[0] [0]  
-----  
conv2d_149 (Conv2D)      (None, 17, 17, 16 122880       mixed5[0] [0]  
-----  
batch_normalization_149 (Batch Normalization) (None, 17, 17, 16 480 conv2d_149[0] [0]  
-----  
activation_149 (Activation) (None, 17, 17, 16 0           batch_normalization_149[0] [  
-----  
conv2d_150 (Conv2D)      (None, 17, 17, 16 179200       activation_149[0] [0]  
-----  
batch_normalization_150 (Batch Normalization) (None, 17, 17, 16 480 conv2d_150[0] [0]  
-----  
activation_150 (Activation) (None, 17, 17, 16 0           batch_normalization_150[0] [  
-----  
conv2d_146 (Conv2D)      (None, 17, 17, 16 122880       mixed5[0] [0]  
-----  
conv2d_151 (Conv2D)      (None, 17, 17, 16 179200       activation_150[0] [0]
```

```
-----  
batch_normalization_146 ( (None, 17, 17, 16 480      conv2d_146[0] [0]  
-----  
batch_normalization_151 ( (None, 17, 17, 16 480      conv2d_151[0] [0]  
-----  
activation_146 (Activatio (None, 17, 17, 16 0      batch_normalization_146[0] [  
-----  
activation_151 (Activatio (None, 17, 17, 16 0      batch_normalization_151[0] [  
-----  
conv2d_147 (Conv2D)      (None, 17, 17, 16 179200 activation_146[0] [0]  
-----  
conv2d_152 (Conv2D)      (None, 17, 17, 16 179200 activation_151[0] [0]  
-----  
batch_normalization_147 ( (None, 17, 17, 16 480      conv2d_147[0] [0]  
-----  
batch_normalization_152 ( (None, 17, 17, 16 480      conv2d_152[0] [0]  
-----  
activation_147 (Activatio (None, 17, 17, 16 0      batch_normalization_147[0] [  
-----  
activation_152 (Activatio (None, 17, 17, 16 0      batch_normalization_152[0] [  
-----  
average_pooling2d_15 (Ave (None, 17, 17, 76 0      mixed5[0] [0]  
-----  
conv2d_145 (Conv2D)      (None, 17, 17, 19 147456 mixed5[0] [0]  
-----  
conv2d_148 (Conv2D)      (None, 17, 17, 19 215040 activation_147[0] [0]  
-----  
conv2d_153 (Conv2D)      (None, 17, 17, 19 215040 activation_152[0] [0]  
-----  
conv2d_154 (Conv2D)      (None, 17, 17, 19 147456 average_pooling2d_15[0] [0]  
-----  
batch_normalization_145 ( (None, 17, 17, 19 576      conv2d_145[0] [0]  
-----  
batch_normalization_148 ( (None, 17, 17, 19 576      conv2d_148[0] [0]  
-----  
batch_normalization_153 ( (None, 17, 17, 19 576      conv2d_153[0] [0]  
-----  
batch_normalization_154 ( (None, 17, 17, 19 576      conv2d_154[0] [0]  
-----  
activation_145 (Activatio (None, 17, 17, 19 0      batch_normalization_145[0] [  
-----  
activation_148 (Activatio (None, 17, 17, 19 0      batch_normalization_148[0] [  
-----  
activation_153 (Activatio (None, 17, 17, 19 0      batch_normalization_153[0] [  
-----  
activation_154 (Activatio (None, 17, 17, 19 0      batch_normalization_154[0] [  
-----  
mixed6 (Concatenate)     (None, 17, 17, 76 0      activation_145[0] [0]
```

```

                                activation_148[0] [0]
                                activation_153[0] [0]
                                activation_154[0] [0]

-----conv2d_159 (Conv2D)      (None, 17, 17, 19 147456    mixed6[0] [0]
-----batch_normalization_159 ( (None, 17, 17, 19 576      conv2d_159[0] [0]
-----activation_159 (Activatio (None, 17, 17, 19 0      batch_normalization_159[0] [
-----conv2d_160 (Conv2D)      (None, 17, 17, 19 258048    activation_159[0] [0]
-----batch_normalization_160 ( (None, 17, 17, 19 576      conv2d_160[0] [0]
-----activation_160 (Activatio (None, 17, 17, 19 0      batch_normalization_160[0] [
-----conv2d_156 (Conv2D)      (None, 17, 17, 19 147456    mixed6[0] [0]
-----conv2d_161 (Conv2D)      (None, 17, 17, 19 258048    activation_160[0] [0]
-----batch_normalization_156 ( (None, 17, 17, 19 576      conv2d_156[0] [0]
-----batch_normalization_161 ( (None, 17, 17, 19 576      conv2d_161[0] [0]
-----activation_156 (Activatio (None, 17, 17, 19 0      batch_normalization_156[0] [
-----activation_161 (Activatio (None, 17, 17, 19 0      batch_normalization_161[0] [
-----conv2d_157 (Conv2D)      (None, 17, 17, 19 258048    activation_156[0] [0]
-----conv2d_162 (Conv2D)      (None, 17, 17, 19 258048    activation_161[0] [0]
-----batch_normalization_157 ( (None, 17, 17, 19 576      conv2d_157[0] [0]
-----batch_normalization_162 ( (None, 17, 17, 19 576      conv2d_162[0] [0]
-----activation_157 (Activatio (None, 17, 17, 19 0      batch_normalization_157[0] [
-----activation_162 (Activatio (None, 17, 17, 19 0      batch_normalization_162[0] [
-----average_pooling2d_16 (Ave (None, 17, 17, 76 0      mixed6[0] [0]
-----conv2d_155 (Conv2D)      (None, 17, 17, 19 147456    mixed6[0] [0]
-----conv2d_158 (Conv2D)      (None, 17, 17, 19 258048    activation_157[0] [0]
-----conv2d_163 (Conv2D)      (None, 17, 17, 19 258048    activation_162[0] [0]
-----
```

```
conv2d_164 (Conv2D)      (None, 17, 17, 19 147456    average_pooling2d_16[0] [0]
-----
batch_normalization_155 ( (None, 17, 17, 19 576      conv2d_155[0] [0]
-----
batch_normalization_158 ( (None, 17, 17, 19 576      conv2d_158[0] [0]
-----
batch_normalization_163 ( (None, 17, 17, 19 576      conv2d_163[0] [0]
-----
batch_normalization_164 ( (None, 17, 17, 19 576      conv2d_164[0] [0]
-----
activation_155 (Activatio (None, 17, 17, 19 0      batch_normalization_155[0] [
-----
activation_158 (Activatio (None, 17, 17, 19 0      batch_normalization_158[0] [
-----
activation_163 (Activatio (None, 17, 17, 19 0      batch_normalization_163[0] [
-----
activation_164 (Activatio (None, 17, 17, 19 0      batch_normalization_164[0] [
-----
mixed7 (Concatenate)     (None, 17, 17, 76 0      activation_155[0] [0]
                                         activation_158[0] [0]
                                         activation_163[0] [0]
                                         activation_164[0] [0]
-----
conv2d_167 (Conv2D)      (None, 17, 17, 19 147456    mixed7[0] [0]
-----
batch_normalization_167 ( (None, 17, 17, 19 576      conv2d_167[0] [0]
-----
activation_167 (Activatio (None, 17, 17, 19 0      batch_normalization_167[0] [
-----
conv2d_168 (Conv2D)      (None, 17, 17, 19 258048    activation_167[0] [0]
-----
batch_normalization_168 ( (None, 17, 17, 19 576      conv2d_168[0] [0]
-----
activation_168 (Activatio (None, 17, 17, 19 0      batch_normalization_168[0] [
-----
conv2d_165 (Conv2D)      (None, 17, 17, 19 147456    mixed7[0] [0]
-----
conv2d_169 (Conv2D)      (None, 17, 17, 19 258048    activation_168[0] [0]
-----
batch_normalization_165 ( (None, 17, 17, 19 576      conv2d_165[0] [0]
-----
batch_normalization_169 ( (None, 17, 17, 19 576      conv2d_169[0] [0]
-----
activation_165 (Activatio (None, 17, 17, 19 0      batch_normalization_165[0] [
-----
activation_169 (Activatio (None, 17, 17, 19 0      batch_normalization_169[0] [
-----
conv2d_166 (Conv2D)      (None, 8, 8, 320) 552960    activation_165[0] [0]
```

```

-----  

conv2d_170 (Conv2D)      (None, 8, 8, 192) 331776 activation_169[0] [0]  

-----  

batch_normalization_166 ( (None, 8, 8, 320) 960 conv2d_166[0] [0]  

-----  

batch_normalization_170 ( (None, 8, 8, 192) 576 conv2d_170[0] [0]  

-----  

activation_166 (Activatio (None, 8, 8, 320) 0 batch_normalization_166[0] [  

-----  

activation_170 (Activatio (None, 8, 8, 192) 0 batch_normalization_170[0] [  

-----  

max_pooling2d_8 (MaxPooli (None, 8, 8, 768) 0 mixed7[0] [0]  

-----  

mixed8 (Concatenate)     (None, 8, 8, 1280 0 activation_166[0] [0]  

activation_170[0] [0]  

max_pooling2d_8[0] [0]  

-----  

conv2d_175 (Conv2D)      (None, 8, 8, 448) 573440 mixed8[0] [0]  

-----  

batch_normalization_175 ( (None, 8, 8, 448) 1344 conv2d_175[0] [0]  

-----  

activation_175 (Activatio (None, 8, 8, 448) 0 batch_normalization_175[0] [  

-----  

conv2d_172 (Conv2D)      (None, 8, 8, 384) 491520 mixed8[0] [0]  

-----  

conv2d_176 (Conv2D)      (None, 8, 8, 384) 1548288 activation_175[0] [0]  

-----  

batch_normalization_172 ( (None, 8, 8, 384) 1152 conv2d_172[0] [0]  

-----  

batch_normalization_176 ( (None, 8, 8, 384) 1152 conv2d_176[0] [0]  

-----  

activation_172 (Activatio (None, 8, 8, 384) 0 batch_normalization_172[0] [  

-----  

activation_176 (Activatio (None, 8, 8, 384) 0 batch_normalization_176[0] [  

-----  

conv2d_173 (Conv2D)      (None, 8, 8, 384) 442368 activation_172[0] [0]  

-----  

conv2d_174 (Conv2D)      (None, 8, 8, 384) 442368 activation_172[0] [0]  

-----  

conv2d_177 (Conv2D)      (None, 8, 8, 384) 442368 activation_176[0] [0]  

-----  

conv2d_178 (Conv2D)      (None, 8, 8, 384) 442368 activation_176[0] [0]  

-----  

average_pooling2d_17 (Ave (None, 8, 8, 1280 0 mixed8[0] [0]  

-----  

conv2d_171 (Conv2D)      (None, 8, 8, 320) 409600 mixed8[0] [0]  

-----  

batch_normalization_173 ( (None, 8, 8, 384) 1152 conv2d_173[0] [0]

```

```
-----  
batch_normalization_174 ( (None, 8, 8, 384) 1152      conv2d_174[0] [0]  
-----  
batch_normalization_177 ( (None, 8, 8, 384) 1152      conv2d_177[0] [0]  
-----  
batch_normalization_178 ( (None, 8, 8, 384) 1152      conv2d_178[0] [0]  
-----  
conv2d_179 (Conv2D)      (None, 8, 8, 192) 245760    average_pooling2d_17[0] [0]  
-----  
batch_normalization_171 ( (None, 8, 8, 320) 960       conv2d_171[0] [0]  
-----  
activation_173 (Activatio (None, 8, 8, 384) 0        batch_normalization_173[0] [  
-----  
activation_174 (Activatio (None, 8, 8, 384) 0        batch_normalization_174[0] [  
-----  
activation_177 (Activatio (None, 8, 8, 384) 0        batch_normalization_177[0] [  
-----  
activation_178 (Activatio (None, 8, 8, 384) 0        batch_normalization_178[0] [  
-----  
batch_normalization_179 ( (None, 8, 8, 192) 576       conv2d_179[0] [0]  
-----  
activation_171 (Activatio (None, 8, 8, 320) 0        batch_normalization_171[0] [  
-----  
mixed9_0 (Concatenate)   (None, 8, 8, 768) 0        activation_173[0] [0]  
                           activation_174[0] [0]  
-----  
concatenate_3 (Concatenat (None, 8, 8, 768) 0        activation_177[0] [0]  
                           activation_178[0] [0]  
-----  
activation_179 (Activatio (None, 8, 8, 192) 0        batch_normalization_179[0] [  
-----  
mixed9 (Concatenate)    (None, 8, 8, 2048 0        activation_171[0] [0]  
                           mixed9_0[0] [0]  
                           concatenate_3[0] [0]  
                           activation_179[0] [0]  
-----  
conv2d_184 (Conv2D)     (None, 8, 8, 448) 917504    mixed9[0] [0]  
-----  
batch_normalization_184 ( (None, 8, 8, 448) 1344      conv2d_184[0] [0]  
-----  
activation_184 (Activatio (None, 8, 8, 448) 0        batch_normalization_184[0] [  
-----  
conv2d_181 (Conv2D)     (None, 8, 8, 384) 786432    mixed9[0] [0]  
-----  
conv2d_185 (Conv2D)     (None, 8, 8, 384) 1548288   activation_184[0] [0]  
-----  
batch_normalization_181 ( (None, 8, 8, 384) 1152      conv2d_181[0] [0]  
-----
```

```

batch_normalization_185 (None, 8, 8, 384) 1152      conv2d_185[0] [0]
-----
activation_181 (Activation) (None, 8, 8, 384) 0      batch_normalization_181[0] [
-----
activation_185 (Activation) (None, 8, 8, 384) 0      batch_normalization_185[0] [
-----
conv2d_182 (Conv2D)      (None, 8, 8, 384) 442368 activation_181[0] [0]
-----
conv2d_183 (Conv2D)      (None, 8, 8, 384) 442368 activation_181[0] [0]
-----
conv2d_186 (Conv2D)      (None, 8, 8, 384) 442368 activation_185[0] [0]
-----
conv2d_187 (Conv2D)      (None, 8, 8, 384) 442368 activation_185[0] [0]
-----
average_pooling2d_18 (Ave) (None, 8, 8, 2048) 0      mixed9[0] [0]
-----
conv2d_180 (Conv2D)      (None, 8, 8, 320) 655360 mixed9[0] [0]
-----
batch_normalization_182 (None, 8, 8, 384) 1152      conv2d_182[0] [0]
-----
batch_normalization_183 (None, 8, 8, 384) 1152      conv2d_183[0] [0]
-----
batch_normalization_186 (None, 8, 8, 384) 1152      conv2d_186[0] [0]
-----
batch_normalization_187 (None, 8, 8, 384) 1152      conv2d_187[0] [0]
-----
conv2d_188 (Conv2D)      (None, 8, 8, 192) 393216 average_pooling2d_18[0] [0]
-----
batch_normalization_180 (None, 8, 8, 320) 960      conv2d_180[0] [0]
-----
activation_182 (Activation) (None, 8, 8, 384) 0      batch_normalization_182[0] [
-----
activation_183 (Activation) (None, 8, 8, 384) 0      batch_normalization_183[0] [
-----
activation_186 (Activation) (None, 8, 8, 384) 0      batch_normalization_186[0] [
-----
activation_187 (Activation) (None, 8, 8, 384) 0      batch_normalization_187[0] [
-----
batch_normalization_188 (None, 8, 8, 192) 576      conv2d_188[0] [0]
-----
activation_180 (Activation) (None, 8, 8, 320) 0      batch_normalization_180[0] [
-----
mixed9_1 (Concatenate)   (None, 8, 8, 768) 0      activation_182[0] [0]
                                         activation_183[0] [0]
-----
concatenate_4 (Concatenate) (None, 8, 8, 768) 0      activation_186[0] [0]
                                         activation_187[0] [0]
-----
```

```

activation_188 (Activation) (None, 8, 8, 192) 0           batch_normalization_188[0] [
-----
mixed10 (Concatenate)      (None, 8, 8, 2048) 0           activation_180[0][0]
                                                               mixed9_1[0][0]
                                                               concatenate_4[0][0]
                                                               activation_188[0][0]
-----
avg_pool (GlobalAveragePo (None, 2048)    0           mixed10[0][0]
-----
predictions (Dense)        (None, 1000)     2049000 avg_pool[0][0]
=====
Total params: 23,851,784
Trainable params: 23,817,352
Non-trainable params: 34,432
-----
```

3 Load model's classes

The predictions of the network can be decoded with *imagenet_decode_predictions*, however we will use the original text file so we know how it is done.

```
In [6]: synsets <- readLines("synset.txt")
length(synsets)
head(synsets)

1000
1. 'n01440764 tench, Tinca tinca' 2. 'n01443537 goldfish, Carassius auratus' 3. 'n01484850 great
white shark, white shark, man-eater, man-eating shark, Carcharodon carcharias' 4. 'n01491361
tiger shark, Galeocerdo cuvieri' 5. 'n01494475 hammerhead, hammerhead shark' 6. 'n01496331
electric ray, crampfish, numbfish, torpedo'
```

4 Image preprocess functions

```
In [71]: # Image preprocess
preproc.image <- function(im, crop = TRUE, dims=3) {
  if (crop) {
    ## Crop the image so it gets same height and width
    shape <- dim(im)
    short.edge <- min(shape[1:2]) # Get the shorter edge from the picture
    # Calculate how much we should crop for each axis
    xx <- floor((shape[1] - short.edge) / 2)
    yy <- floor((shape[2] - short.edge) / 2)
    im <- crop.borders(im, xx, yy) # Cropped image
  }
}
```

```

# Resize to 299 x 299, needed by input of the model.
resized <- resize(im, 299, 299)
# Convert to array (x, y, channel)
arr <- (as.array(resized) - 0.5)*2 # Pixels between -1 and 1 (This is the normalization)
dim(arr) <- c(299, 299, 3)

# Reshape to format needed by the network (num, width, height, channel)
dim(arr) <- c(1,299, 299, 3)
return(arr)
}

```

In [72]: # Result printing

```

printClassRank <- function(prob, labels, nRes = 10) {
  nRes <- min(nRes, length(labels))
  o <- order(prob, decreasing=TRUE)
  res <- data.frame(class=synsets[o], probability=prob[o])
  head(res, n = nRes)
}

```

5 Image classification

In [73]: # Modifying plot area so that we can see the original photo and the preprocessed
photo

```

oldpar <- par() # We can save the old configuration using empty par()
par(mfrow=c(1,2))

```

5.1 Starters: Take an image from imageR package - Give me macaws!

In [74]: im <- load.image(system.file("extdata/parrots.png", package="imager"))
plot(im)



In [75]: `dim(im)`

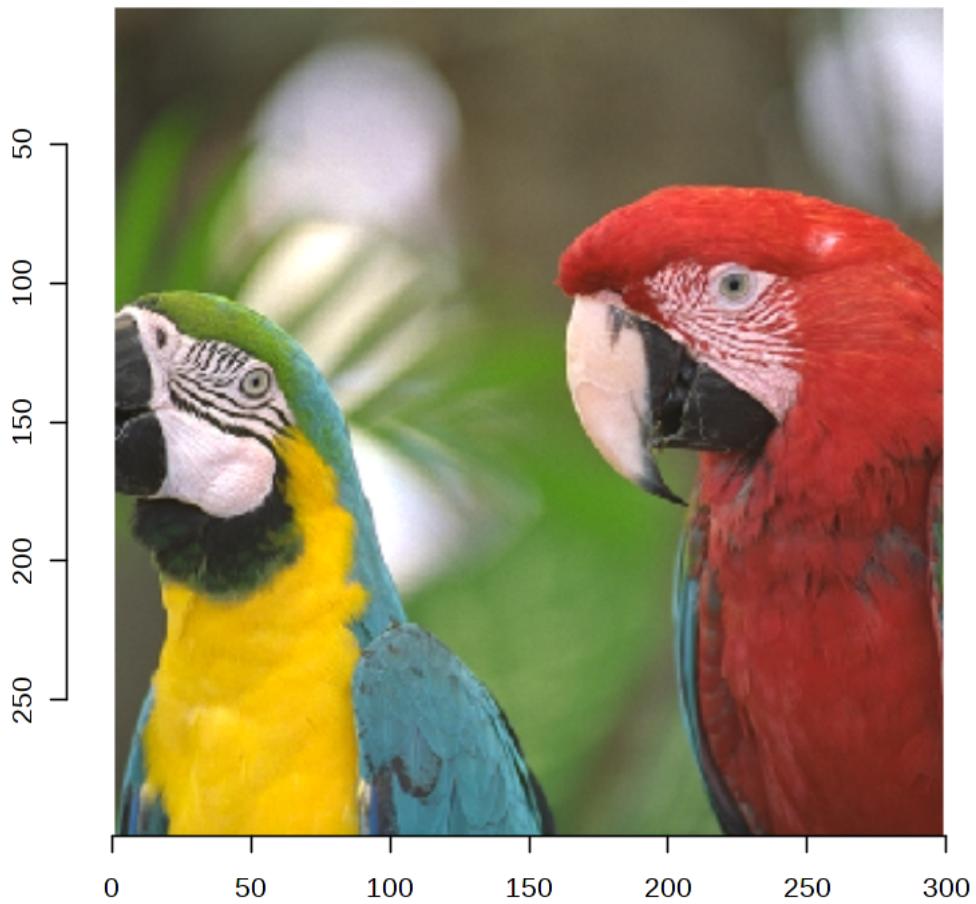
1. 768 2. 512 3. 1 4. 3

768x512 resolution with 3 channels (RGB), but this network was trained with images of size 224x224, we need to resize them.

```
In [93]: preproc <- preproc.image(im)
          plot(as.cimg(preproc[, , ]))
          dim(preproc)
```

Warning message in `as.cimg.array(preproc[, ,])`:
"Assuming third dimension corresponds to colour"

1. 1 2. 299 3. 299 4. 3



We can observe that the image has been reduced, now has height = width and the image is cropped. Values are now between -1 and 1.

```
In [94]: str(preproc)
```

```
num [1, 1:299, 1:299, 1:3] -0.333 -0.349 -0.349 -0.349 -0.357 ...
```

Predict the parrots! In other words, get the class probabilities

```
In [95]: prob <- predict(model, preproc)  
dim(prob)
```

```
1. 1 2. 1000
```

We get 1000 probabilities, one for each class. Now we check which are the most representative.

```
In [96]: printClassRank(prob, synsets)
```

	class	probability
	n01818515 macaw	0.6544093490
	n01820546 lorikeet	0.0186850950
	n01817953 African grey, African gray, Psittacus erithacus	0.0049923691
	n02007558 flamingo	0.0019017688
	n01843383 toucan	0.0017725008
	n01819313 sulphur-crested cockatoo, Kakatoe galerita, Cacatua galerita	0.0016499582
	n03127747 crash helmet	0.0011452826
	n02128757 snow leopard, ounce, Panthera uncia	0.0010587695
	n02012849 crane	0.0009335988
	n03379051 football helmet	0.0009322278

It's actually not a parrot but a Macaw and it gets correctly predicted!

5.2 Now with something completely different: A laptop with Alpha Channel

```
In [97]: im2 <- load.image("images/laptop.png")
plot(im2)
dim(im2)
```

1. 573 2. 430 3. 1 4. 4



572x430 with 4 channels... something is wrong. PNG format allows the image to have transparency using the well-known alpha channel. This channel marks how much transparency we need to use for each pixel. We can remove it using `rm.alpha` function.

```
In [98]: im2 <- rm.alpha(im2)
plot(im2)
dim(im2)
```

1. 573 2. 430 3. 1 4. 3



Now we have 3 channels, we can proceed as before.

```
In [100]: preproc <- preproc.image(im2)
          plot(as.cimg(preproc[, , ]))
```

```
Warning message in as.cimg.array(preproc[, , ]):
"Assuming third dimension corresponds to colour"
```



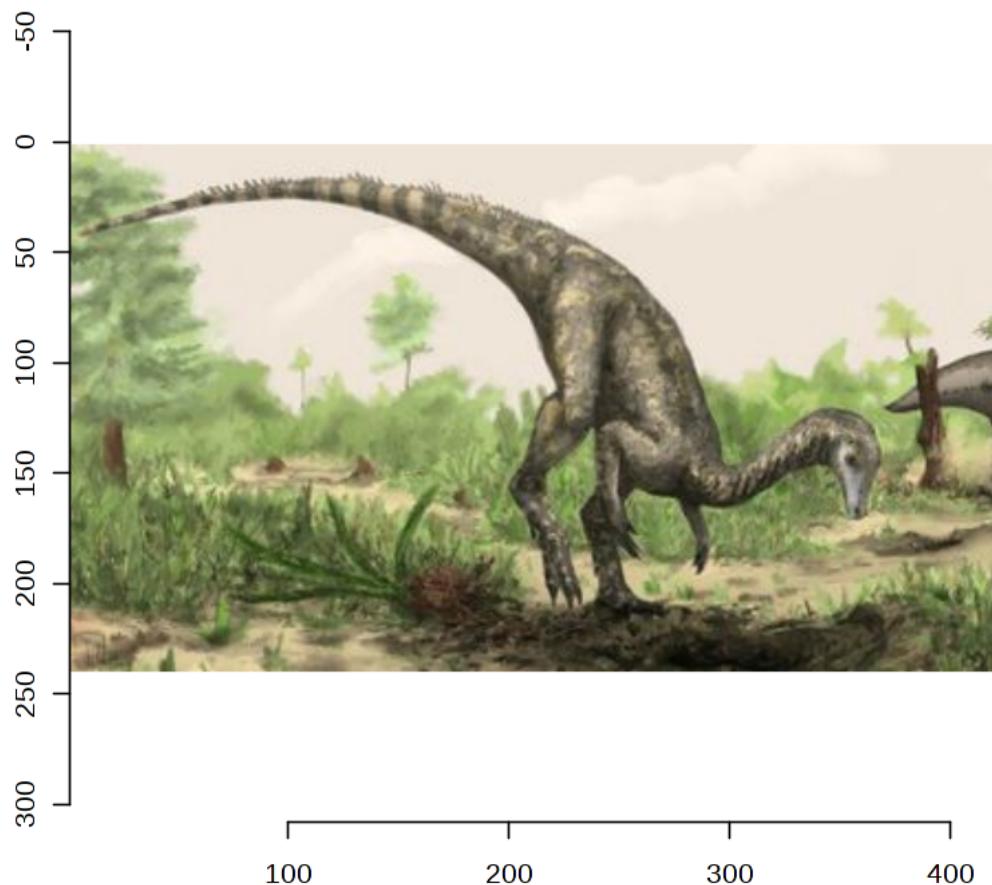
```
In [101]: prob <- predict(model, preproc)
printClassRank(prob, synsets)
```

class	probability
n03782006 monitor	0.660042644
n04152593 screen, CRT screen	0.155979887
n03180011 desktop computer	0.143689096
n03642806 laptop, laptop computer	0.008392184
n03832673 notebook, notebook computer	0.005269927
n03529860 home theater, home theatre	0.001873044
n03085013 computer keyboard, keypad	0.001839006
n03793489 mouse, computer mouse	0.001700508
n04149813 scoreboard	0.001240686
n03857828 oscilloscope, scope, cathode-ray oscilloscope, CRO	0.000899285

Monitor, screen, desktop computer, laptop. Not bad.

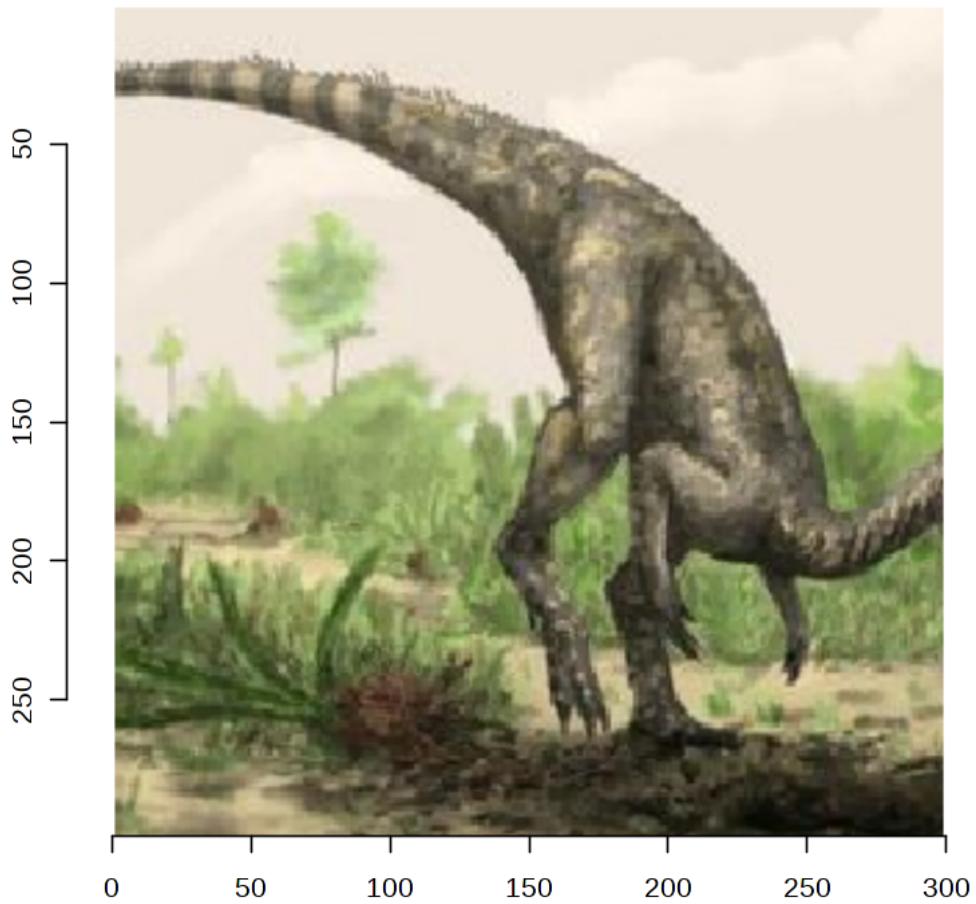
5.3 Picture proportion is important: Is the network aware of dinosaurs?

```
In [102]: im3 <- load.image("images/dinosaur.jpg")
plot(im3)
```



```
In [103]: preproc <- preproc.image(im3)
plot(as.cimg(preproc[, , ]))
prob <- predict(model, preproc)
```

```
Warning message in as.cimg.array(preproc[, , ]):
"Assuming third dimension corresponds to colour"
```



The head gets cut, does it matter?

In [104]: `printClassRank(prob, synsets)`

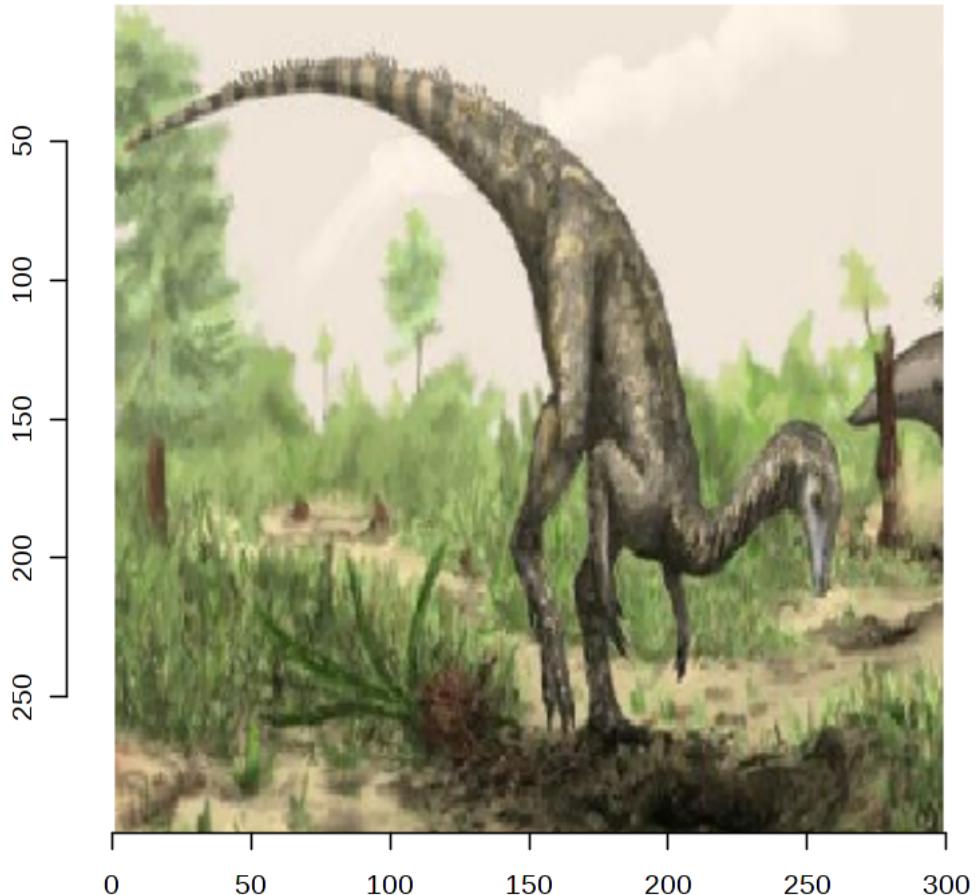
class	probability
n01688243 frilled lizard, Chlamydosaurus kingi	0.326780558
n01694178 African chameleon, Chamaeleo chamaeleon	0.299931347
n01675722 banded gecko	0.068925969
n02457408 three-toed sloth, ai, Bradypus tridactylus	0.023125000
n01644900 tailed frog, bell toad, ribbed toad, tailed toad, Ascaphus trui	0.022452381
n01687978 agama	0.010997850
n02879718 bow	0.007947814
n01689811 alligator lizard	0.005979110
n02500267 indri, indris, Indri indri, Indri brevicaudatus	0.005354682
n01630670 common newt, Triturus vulgaris	0.005207573

African Chamaleon. Almost!

What happens if we resize without cropping the image so that the head is preserved?

```
In [106]: preproc <- preproc.image(im3, crop=FALSE)
          plot(as.cimg(preproc[, , ]))
          prob <- predict(model, preproc)
```

```
Warning message in as.cimg.array(preproc[, , ]):
"Assuming third dimension corresponds to colour"
```



```
In [107]: # Which class is the most representative
```

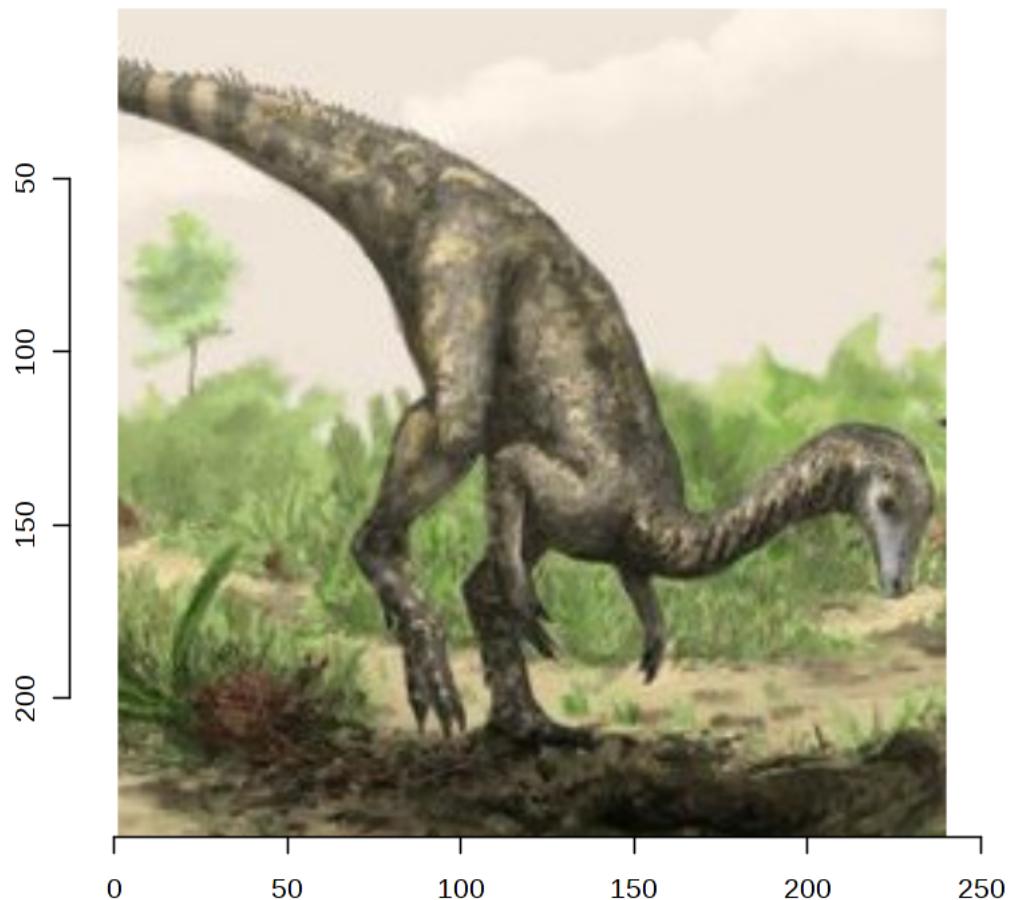
```
printClassRank(prob, synsets)
```

```
# Frilled lizard. The prediction changed, so the proportions of the image are import
```

	class	probability
n01688243	frilled lizard, Chlamydosaurus kingi	0.720620751
n01675722	banded gecko	0.104003742
n01694178	African chameleon, Chamaeleo chamaeleon	0.030216876
n01677366	common iguana, iguana, Iguana iguana	0.017132044
n01689811	alligator lizard	0.011024418
n01687978	agama	0.006645061
n02879718	bow	0.005944965
n01644900	tailed frog, bell toad, ribbed toad, tailed toad, Ascaphus trui	0.004173551
n01698640	American alligator, Alligator mississippiensis	0.003145227
n01682714	American chameleon, anole, Anolis carolinensis	0.003021359

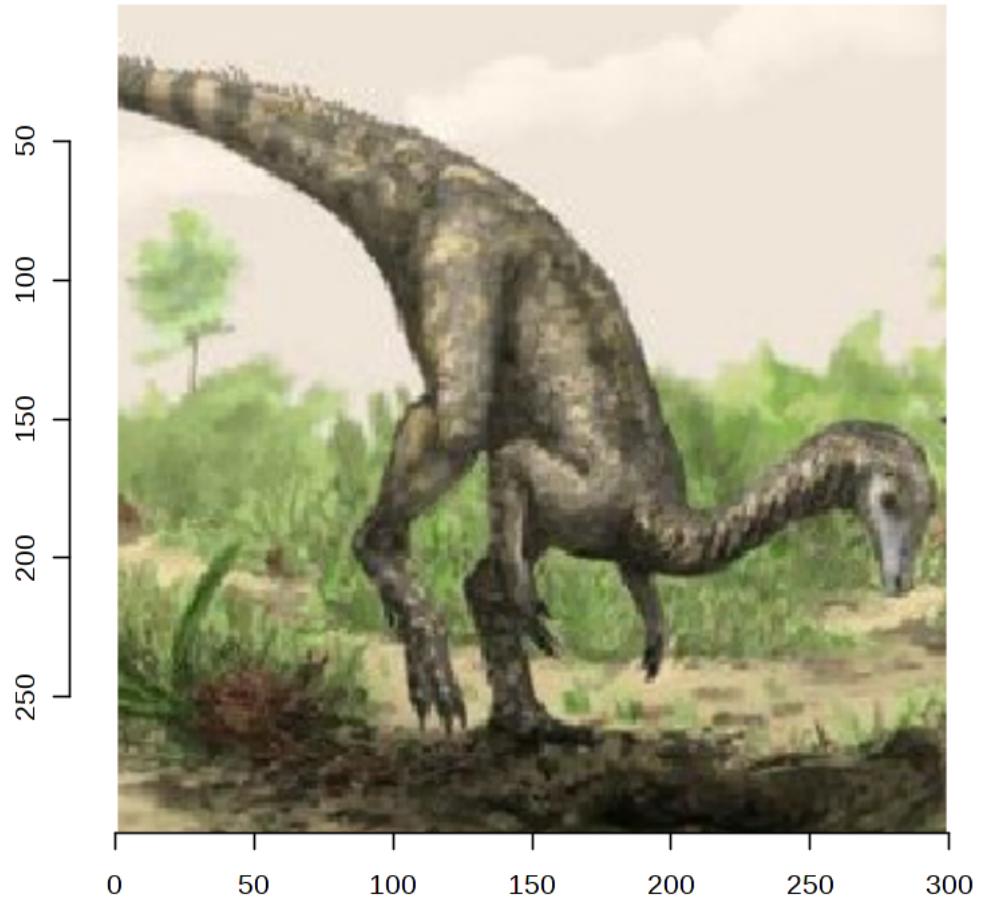
What if we center it manually?

```
In [108]: im3.2 <- load.image("images/dinosaurSquare.jpg")
plot(im3.2)
```



```
In [109]: preproc <- preproc.image(im3.2)
plot(as.cimg(preproc[1,,]))
prob <- predict(model, preproc)

Warning message in as.cimg.array(preproc[1, , ]):
"Assuming third dimension corresponds to colour"
```



```
In [59]: printClassRank(prob, synsets)
```

	class	probability
n01694178	African chameleon, Chamaeleo chamaeleon	0.270068109
n01688243	frilled lizard, Chlamydosaurus kingi	0.157671139
n01675722	banded gecko	0.083946444
n01687978	agama	0.021566324
n02457408	three-toed sloth, ai, Bradypus tridactylus	0.019916981
n01682714	American chameleon, anole, Anolis carolinensis	0.014364913
n01644900	tailed frog, bell toad, ribbed toad, tailed toad, Ascaphus trui	0.012171571
n02500267	indri, indris, Indri indri, Indri brevicaudatus	0.012092926
n01689811	alligator lizard	0.009442206
n01677366	common iguana, iguana, Iguana iguana	0.009255226
African Chamaleon still is the most representative		

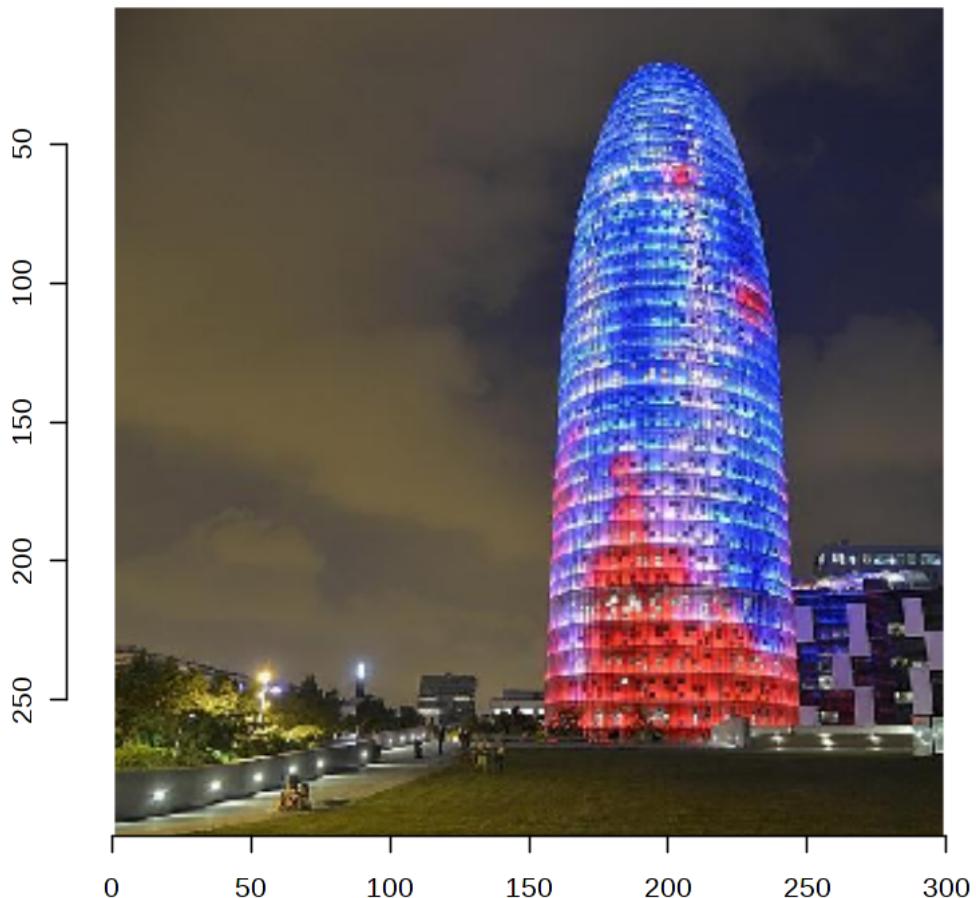
5.4 Can the network identify Agbar Tower?

```
In [111]: im4 <- load.image("images/agbar.jpg")
plot(im4)
```



```
In [112]: preproc <- preproc.image(im4)
plot(as.cimg(preproc[1,,,]))
prob <- predict(model, preproc)
```

Warning message in `as.cimg.array`(`preproc[1, , ,]`):
"Assuming third dimension corresponds to colour"



```
In [113]: printClassRank(prob, synsets)
```

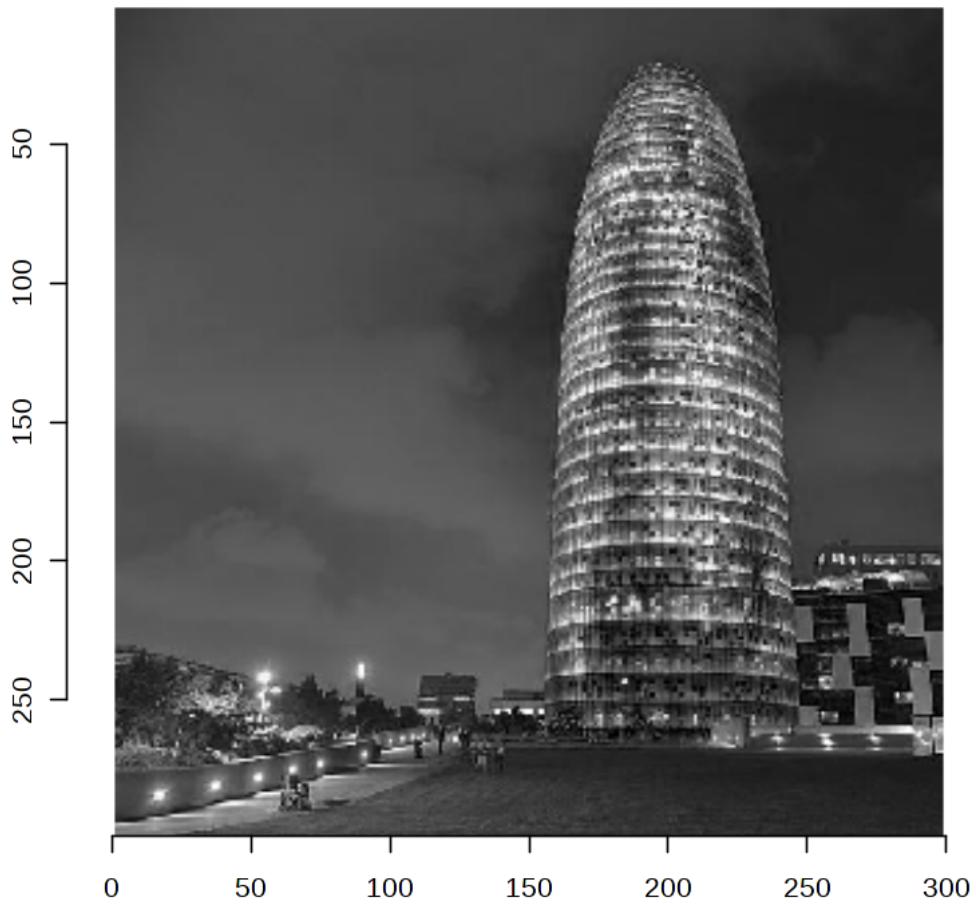
class	probability
n04277352 spindle	0.148714066
n04254777 sock	0.074724309
n04067472 reel	0.028994642
n04296562 stage	0.027421901
n02804610 bassoon	0.020592168
n03838899 oboe, hautboy, hautbois	0.013446912
n03976657 pole	0.010576974
n04141076 sax, saxophone	0.010381553
n04599235 wool, woolen, woollen	0.009507323
n03249569 drum, membranophone, tympan	0.008706241

Spindle, sock... Almost. The network is not quite sure about what Agbar tower is. We should ignore the classification, as the higher probability is very small. Maybe the colors are fooling the network, let's try without color.

```
In [125]: preproc <- preproc.image(im4)
          preproc <- preproc[, , c(2, 2, 2), drop=F] # Grayscale (setting all the channels to the same value)
          plot(as.cimg(preproc[, , ]))
          prob <- predict(model, preproc[, , c(1, 1, 1), drop=F])
          printClassRank(prob, synsets)
```

Warning message in as.cimg.array(preproc[, ,]):
 "Assuming third dimension corresponds to colour"

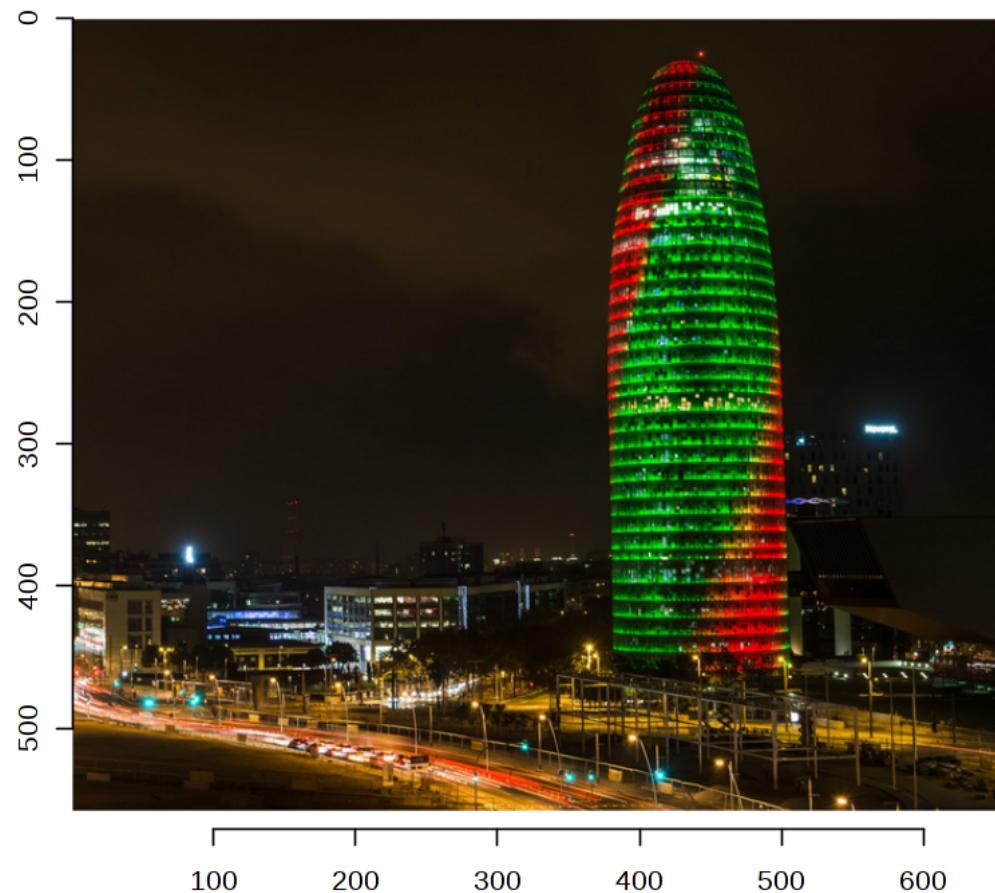
	class	probability
	n04141076 sax, saxophone	0.13501789
	n03495258 harp	0.13326302
n02788148	bannister, banister, balustrade, balusters, handrail	0.10081862
	n03065424 coil, spiral, volute, whorl, helix	0.07746976
	n02804610 bassoon	0.03745870
	n03838899 oboe, hautboy, hautbois	0.03606398
	n04153751 screw	0.02480378
	n02787622 banjo	0.01503362
	n04487394 trombone	0.01327600
n03110669	cornet, horn, trumpet, trump	0.01120354



It's not the color, it gets classified as a saxophone.

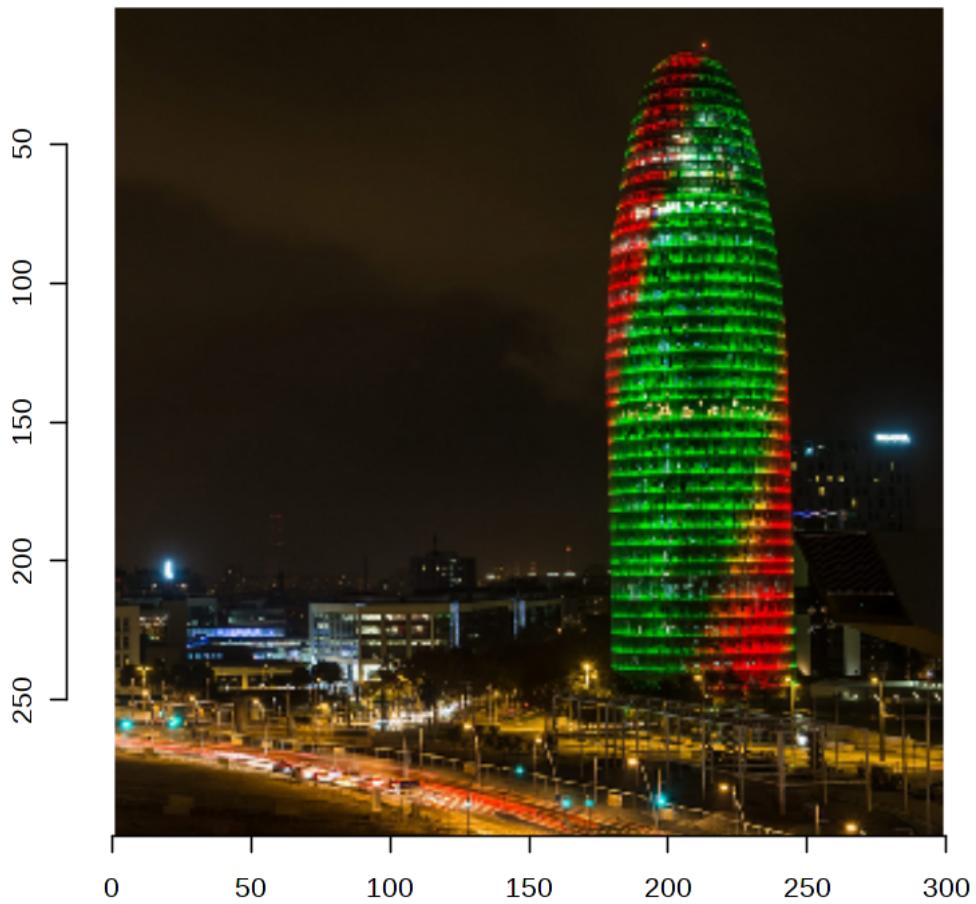
5.5 Agbar 2: Let's try with another photo

```
In [126]: im5 <- load.image("images/Agbar2.png")
plot(im5)
```



```
In [127]: preproc <- preproc.image(im5)
plot(as.cimg(preproc[, , ]))
prob <- predict(model, preproc)
```

Warning message in `as.cimg.array`(`preproc[1, ,]`):
"Assuming third dimension corresponds to colour"



```
In [128]: printClassRank(prob, synsets)
```

class	probability
n03637318 lampshade, lamp shade	0.053805571
n04366367 suspension bridge	0.025773792
n04141076 sax, saxophone	0.017962690
n03388043 fountain	0.014260580
n01930112 nematode, nematode worm, roundworm	0.012751474
n04296562 stage	0.008149481
n04418357 theater curtain, theatre curtain	0.008111350
n03032252 cinema, movie theater, movie theatre, movie house, picture palace	0.007397199
n02782093 balloon	0.007117193
n04254777 sock	0.006699307

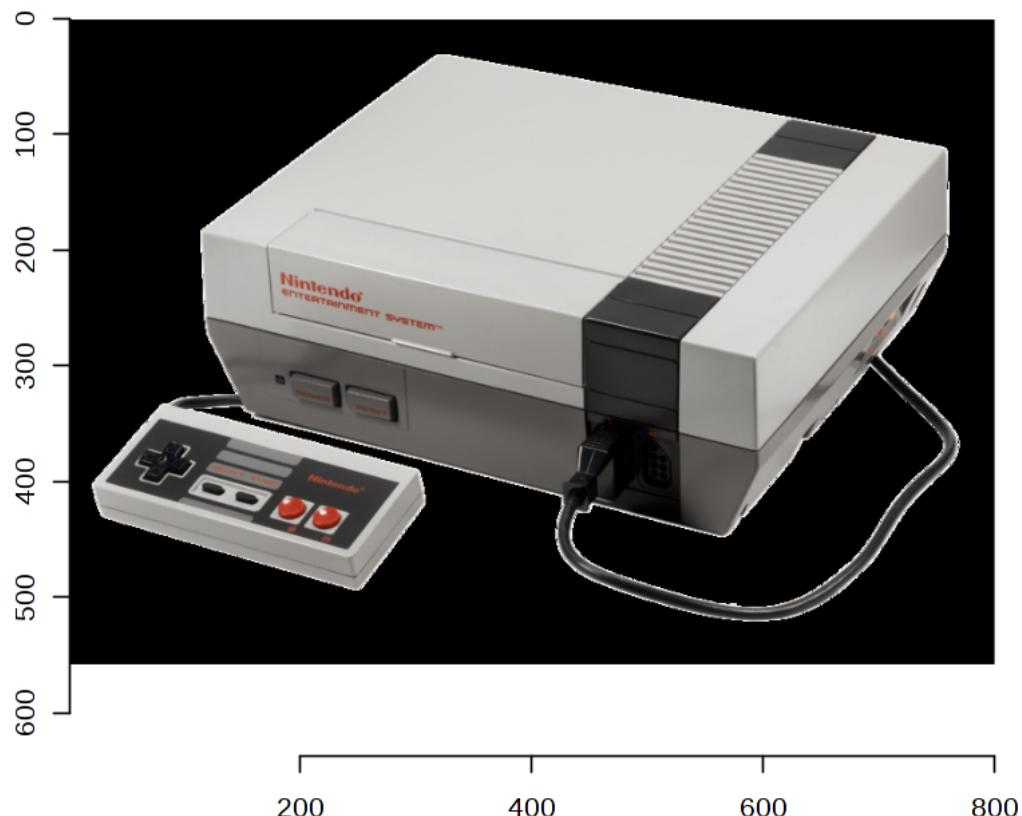
The network is still not clear about what is that photo. Probably the biggest lampshade on planet Earth.

5.6 Identifying by aesthetics: You may know what is this one

```
In [130]: im6 <- load.image("images/NES.png")
dim(im6)
# Notice that this image has 4 channels!
plot(im6)
```

Warning message in `readfun(f, ...)`:
"libpng warning: iCCP: known incorrect sRGB profile"

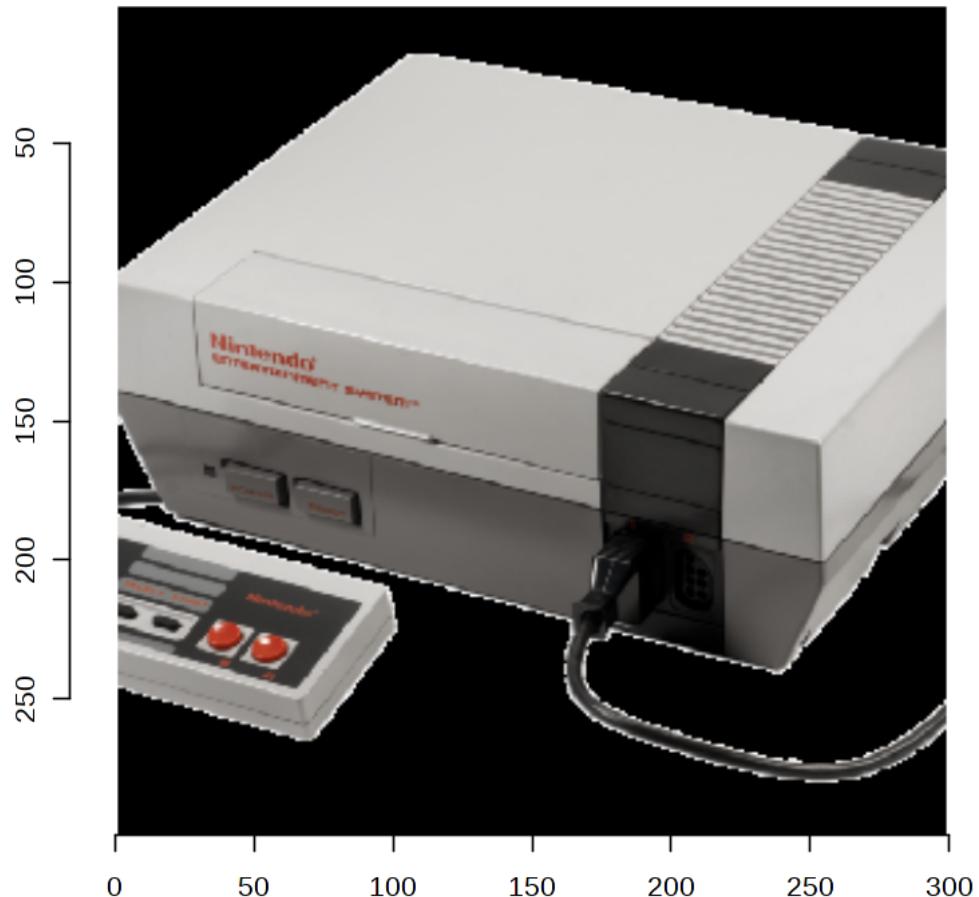
1. 800 2. 558 3. 1 4. 4



```
In [131]: # Remove Alpha Channel  
im6 <- rm.alpha(im6)
```

```
In [132]: preproc <- preproc.image(im6)  
plot(as.cimg(preproc[, , ]))  
prob <- predict(model, preproc)
```

Warning message in as.cimg.array(preproc[, ,]):
"Assuming third dimension corresponds to colour"



```
In [69]: printClassRank(prob, synsets)
```

	class	probability
	n03777754 modem	0.19551541
	n03782006 monitor	0.10794843
	n04152593 screen, CRT screen	0.06946526
n03976467	Polaroid camera, Polaroid Land camera	0.05962937
	n04404412 television, television system	0.05094223
	n03924679 photocopier	0.04266527
	n02979186 cassette player	0.03596348
	n04004767 printer	0.03188616
	n03297495 espresso maker	0.03042920
	n04009552 projector	0.01736374

Modem, cassette player, CRT screen... An hypothesis on this is that it is recognizing the aesthetics of that time this console was built.