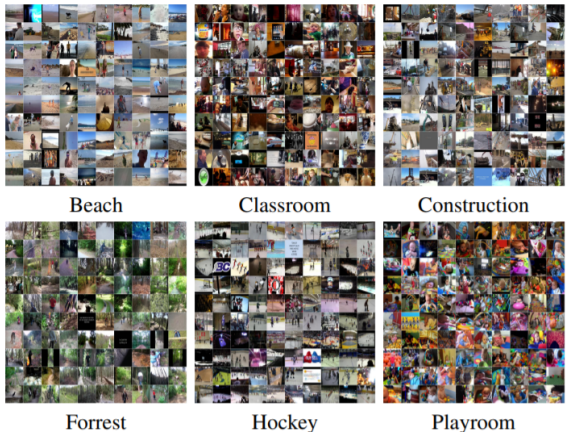
SoundNet: Learning Sound Representations from Unlabeled Video

ARIAS Camila
IBARRA Kevin

Problem



- There are a lot of works in object recognition, speech, and machine translation using labeled dataset. It is not the same corresponding progress in natural sound understanding.
- Other studies are focused on features such as spectrograms and MFCC, on another hand, Soundnet is focused on the natural sounds.
- To handle the problem with labels, the authors capitalize on the natural relation between pictures and sounds in videos using a student-teacher training procedure.

How to transfer?

An unsupervised deep convolutional network

- learns directly from the **raw audio** waveform (it means there are not pre-processing in input data)
- Trained by video unlabeled as a bridge.
- Without truth sound labels.

- Transfer knowledge (Teacher - Student perspective)

Another important aspects:

- Dataset of 2M videos from Flickr.
- Reduce sample rate to 22 kHz and single channel.

Architecture

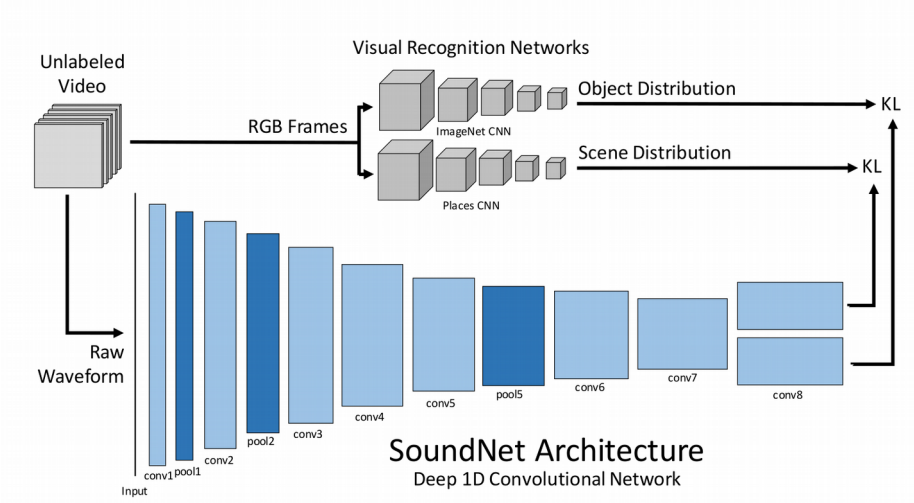
SoundNet is a deep convolutional network

1. Fully convolutional network: One dimensional convolutional layer + ReLU activation layer

Why does it use convolutional layer into sound data?

translation invariant reducing the number of parameters and they allow **stack layers,** useful to detect higher-level concept.

2. Pooling Layers
   To down-sample variable length inputs.



Math in training

In training phase was used video, but the aim is to recognize sounds. For this reason, in the compiled model videos are not the input.

- Waveform
  - $X_i \in R^D$
- Video Images
  - $Y_i \in R^{3xTxWxH}$, where W: width, H: height T: # of Samples
  - For $1 \leq i \leq N$

The system use $g_k(y_i)$, learned with the images to teach to $f_k(x_i)$, k are the objects the knowledge transfer.

$\min_\theta \sum_{k=1}^{K} \sum_{i=1}^{N} D_{KL}\left(g_k(y_i) || f_k(x_i;\theta)\right)$, where $D_{KL}\,(P||Q)$ is a probability distribution (Information divergence, Kullback-Leibler divergence)

The authors chose KL-divergence because is differentiable, then it is possible optimize it using back-propagation and stochastic gradient descent.

Implementation

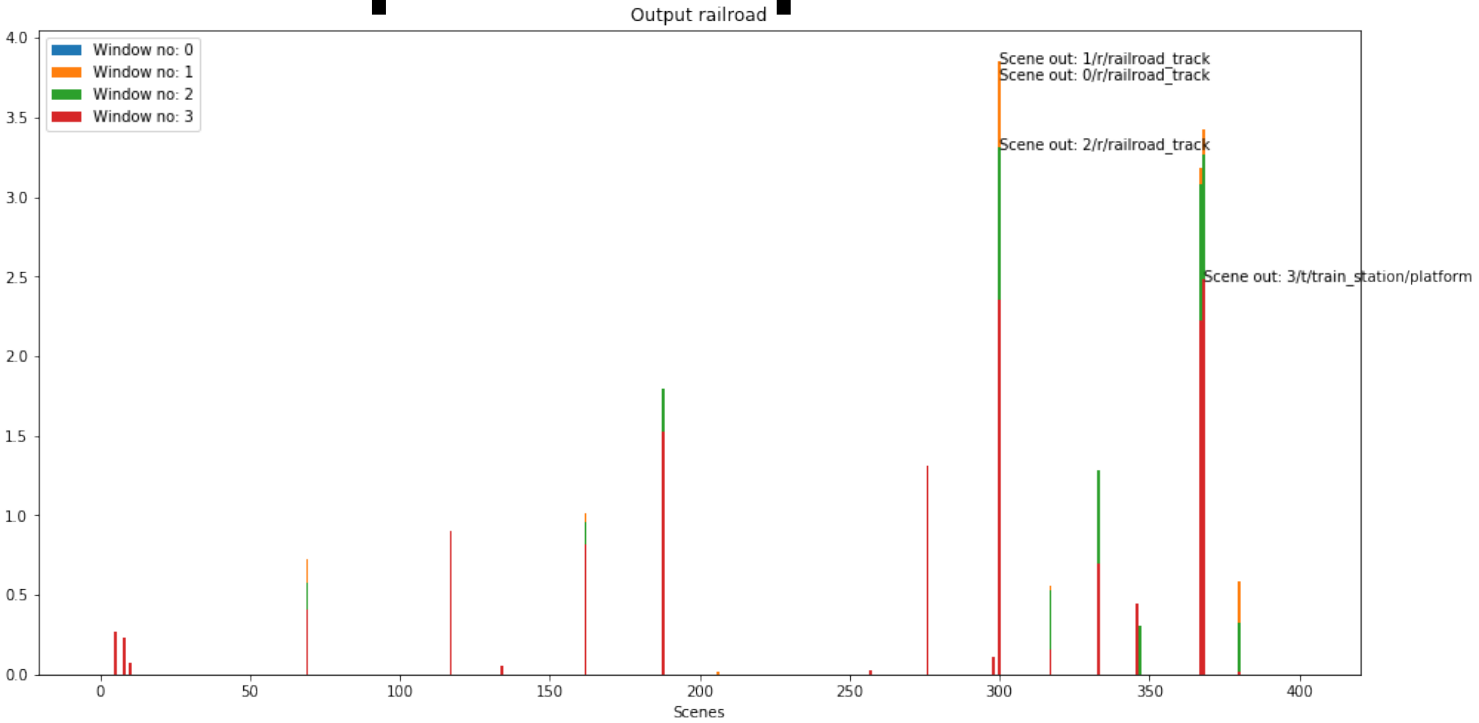
The configuration of the layers

Model pre-trained

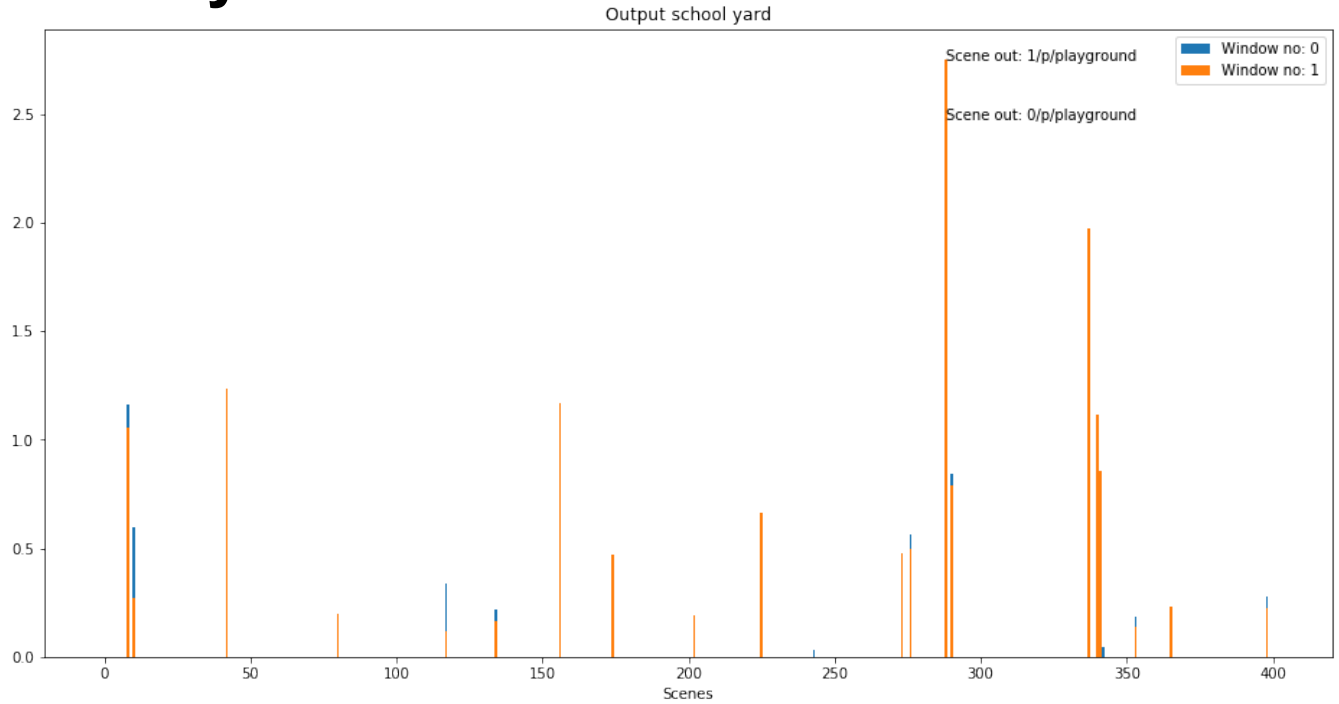
With build_model is built SoundNet according to the structure defined by the authors. This model has as input the audio raw waveform and two outputs: scenes and objects distribution.

Layer	conv1	pool1	conv2	pool2	conv3	conv4	conv5	pool5	conv6	conv7	conv8
Dim.	220,050	27,506	13,782	1,722	862	432	217	54	28	15	4
# of Filters	16	16	32	32	64	128	256	256	512	1024	401
Filter Size	64	8	32	8	16	8	4	4	4	4	8
Stride	2	1	2	1	2	2	2	1	2	2	2
Padding	32	0	16	0	8	4	2	0	2	2	0

Input - Output SoundNet



Experiment with another audio: school yard

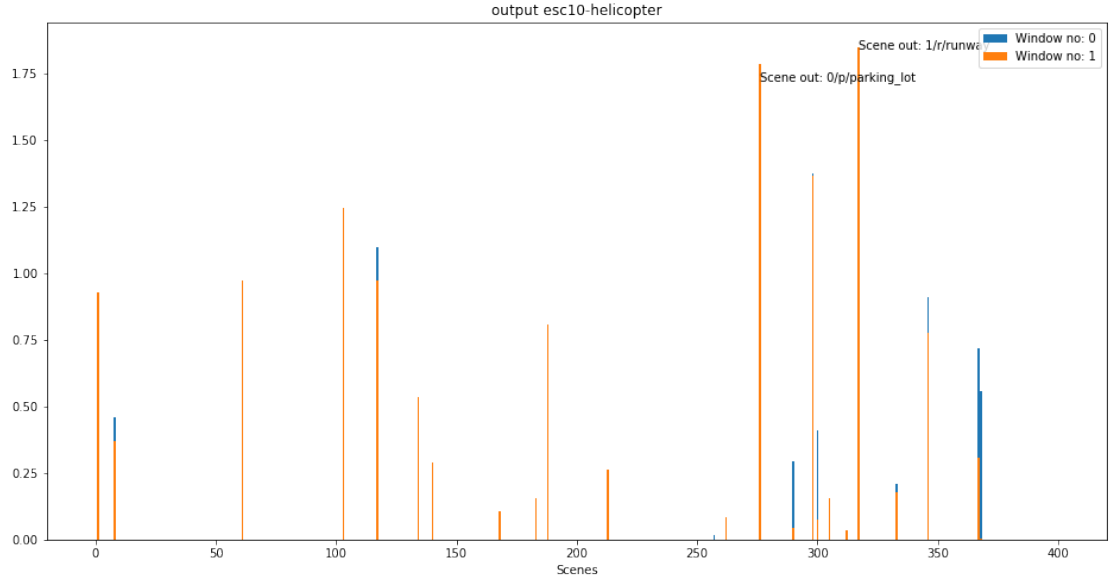


Minimum size

The number of prediction vectors (output-size) depends on the size of the input. It means input must have a minimum size to allow the model predict some output.

Compilation error: the audio has a duration of 5s (110250 samples / 22050 samples per second). The model does not have the minimum number of required samples and is reflected in the Keras kernel. It turns off because of this error.

```
>datos = np.asarray([data[5],data[5],data[5]]).reshape(1,-1,1)
```



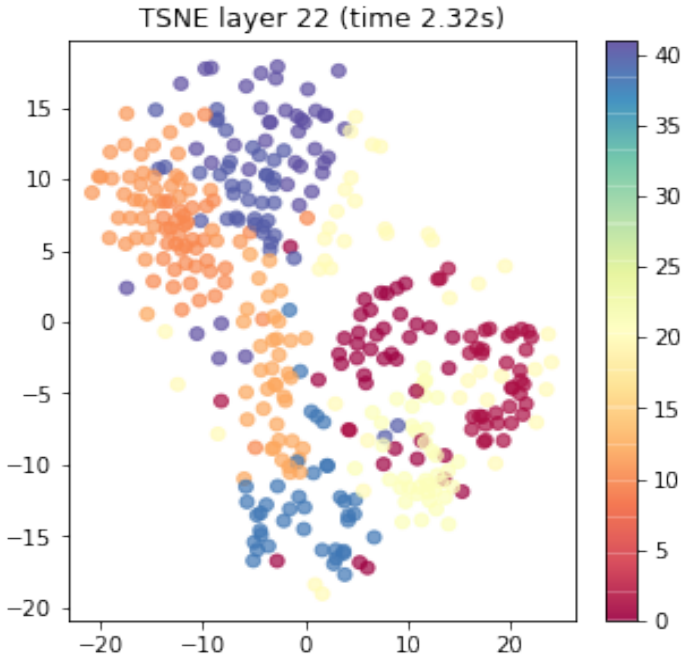
Own experiments

In order to evaluate the model behavior we use the dataset **ESC-10.** It is a subset of ESC-50 which consists of 10 classes (dog bark, rain, sea waves, baby cry, clock tic, person sneeze, helicopter, chainsaw, rooster, and fire cracking).

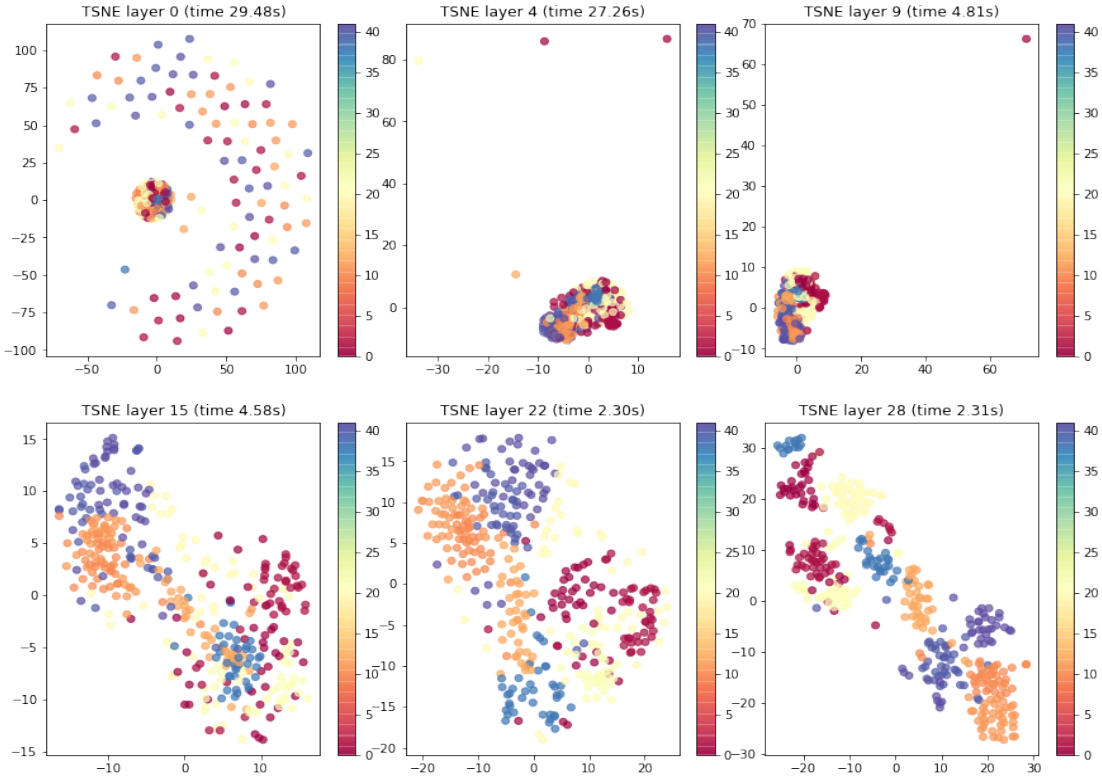
High level features and sound classifier

1) To find features in hidden layers in order to transfer learning, that will be presented below via TSNE graphs.
2) To design and to train a new classifier using as input the output of a hidden layer (pool5) which has been chosen due to the lower number of parameter in the network.

How does the output look like in layer Pool5?



Neuronal networks: find relationships

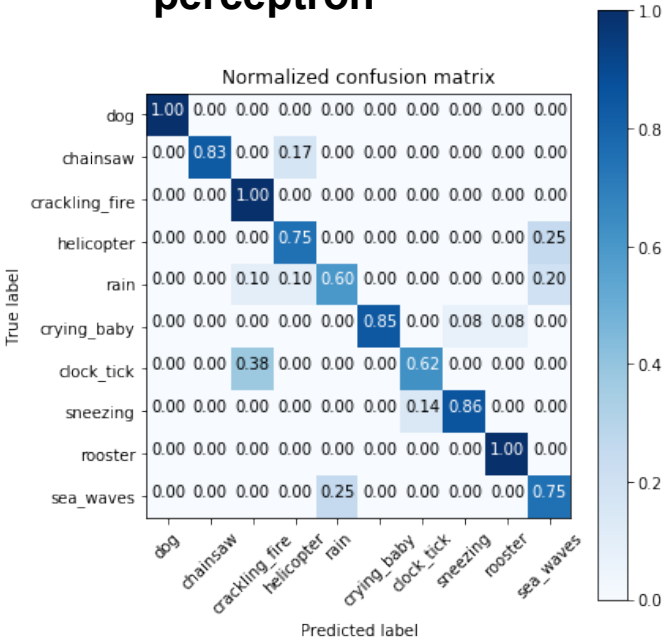


Classifier from hidden layers

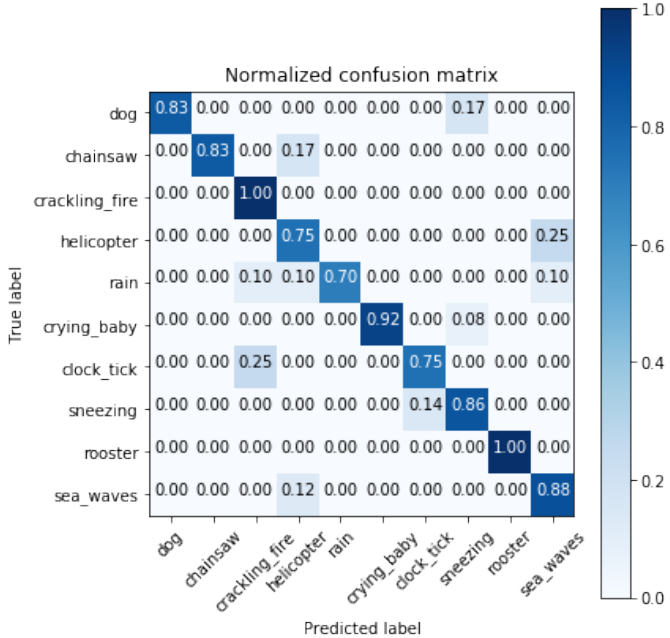
Training for two type of audio classifiers which exploit the internal representation obtained in the 22nd layer of SoundNet in order to classifier other types of sounds.

1. Single-layer perceptron (classifier with Softmax)
2. Linear SVM

Single-layer perceptron



Linear SVM



Dépot github

https://github.com/camila-ud/SoundNet-keras