DEEP HIDDEN ANALYSIS: A STATISTICAL FRAMEWORK TO PRUNE FEATURE MAPS

Arshdeep Singh, Padmanabhan Rajan, Arnav Bhavsar

School of Computing and Electrical Engineering
Indian Institute of Technology, Mandi
Email: d16006@students.iitmandi.ac.in, padman@iitmandi.ac.in, arnav@iitmandi.ac.in

ABSTRACT

In this paper, we propose a statistical framework to prune feature maps in 1-D deep convolutional networks. SoundNet is a pre-trained deep convolutional network that accepts raw audio samples as input. The feature maps generated at various layers of SoundNet have redundancy, which can be identified by statistical analysis. These redundant feature maps can be pruned from the network with a very minor reduction in the capability of the network. The advantage of pruning feature maps, is that computational complexity can be reduced in the context of using an ensemble of classifiers on the layers of SoundNet. Our experiments on acoustic scene classification demonstrate that ignoring 89% of feature maps reduces the performance by less than 3% with 18% reduction in computational complexity.

Index Terms— Pruning, SoundNet, statistical analysis, acoustic scene classification.

1. INTRODUCTION

Recently deep convolutional networks (CNN) have shown performance equivalent to that of a dermatologist in processing images for detecting skin cancer [1]. Several networks have been successfully utilised in various computer vision and image analysis tasks, whereas their use in processing raw audio signals are only coming up. The lack of large labeled datasets as in the vision community has held back the creation of large-scale networks for the processing of raw audio signals [2, 3]. Typically, most networks processing audio signals utilise the spectrogram matrix (or its representation in image form) [4, 5]. Recently, the deep network SoundNet has been created for analysis of raw audio waveforms [3]. SoundNet showed good performance in various audio processing tasks [6, 7].

An issue which has been studied recently is that of *pruning* large-scale networks, again mostly in computer vision. Complex networks typically have thousands of parameters, some of which can be discarded. For example, the study in [8] proposed an energy-driven procedure to prune weights layerwise in networks like AlexNet and GoogLeNet. The study in [9] measured the importance of units in the second-to-last layer before the classification layer, and has utilised this information during training. Units with lesser importance are pruned without affecting performance significantly. In [10], the authors reduced storage requirements by pruning, quantization and Huffman coding of the weights.

However, the shortcomings of most of these methods are that it produces irregular networks which may not have significant reduction in computational cost owing to removal of weights only [11]. The irregular networks require extra efforts to improve convergence speed and performance. In addition, due to irregular connections, these networks have difficulty on parallel computation too [12].

In this regard, [13] reduces the computational cost of CNNs by removing the filters with smallest sum of weights together with their connecting feature maps. The study in [14] considers filter pruning as an optimization problem, and eliminates filters based on statistics computed from its next layer. The study in [15] reduces the number of input channels for each filter and the number of filters in a given layer based on low variance-criterion. These filter pruning methods reduces computational complexity significantly. However, the pruned networks have to be fine-tuned to maintain the performance. Hence, these pruning strategies can be considered data dependent.

Some recent studies have attempted to improve performance by not just utilising the features from last layer being fed in the classifier, but also using intermediate layers. For example, the studies [16, 17] seek to combine features from multiple layers in fast R-CNN and VGG16 before making a prediction for an object detection task. The study in [18] developed an ensemble of classifiers utilizing information from intermediate layers of SoundNet.

In this work, we propose a pruning framework to identify redundant feature maps from various layers of a pre-trained network (henceforth called the baseline network), for compact layer-wise feature representation. Our hypothesis is that, some feature maps give similar responses to inputs of various classes, thereby reducing their utility in discrimination. We call such feature maps *redundant.* We identify redundant feature maps using three statistical measures. In the context of utilizing an ensemble of classifiers for feature maps from various layers, redundant feature maps can be ignored. This results in a reduction of computation during inference of the pre-trained model, and during the training phase of the classifier ensemble. Moreover, since the baseline network is essentially unchanged, there is no need for fine-tuning the pruned network. This is especially important when there is a lack of large amounts of data.

Our framework identifies the redundant feature maps for different layers independently using significance-based, entropy based, angle deviation based measures across different classes. Since these measures are estimated collectively using examples from various classes, between-class information is incorporated for pruning. The key advantages and major contributions of this paper can be summarized as follows:

- Our proposed statistical framework can be applied to pre-trained networks, and simultaneously reduces the dimensionality and computational complexity by ignoring redundant feature maps. The proposed framework also preserves the architecture of the baseline network.

- In experiments on SoundNet, the proposed method utilizes the hidden information and reduces computational complexity by approx. 18% using only 11% of feature maps with a degradation of performance less than 2.80% on two acoustic scene classification datasets.

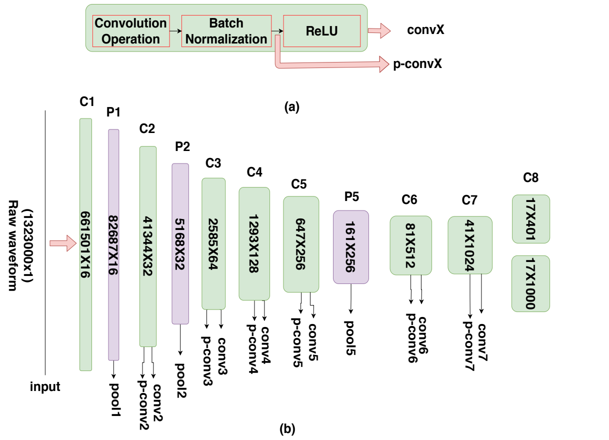


Fig. 1. SoundNet architecture [3]. (a) convolution layer architecture, convX denotes the output of Xth convolution layer output and p-convX denotes the output of batch normalization layer. (b) 8-layer architecture with CX and PX as Xth convolution and pooling layers respectively.

The rest of this paper is organized as follows. In section 2, we briefly introduce SoundNet and illustrate how redundancy of feature maps can be determined. Subsequently, the proposed method is described in section 3. Performance evaluation and conclusions are included in section 4 and 5 respectively.

2. BACKGROUND

2.1. A brief on SoundNet

SoundNet [3] is a 1D convolution neural network (CNN) trained on unlabeled raw audio signals using transfer learning from unlabeled video. The 8-layer SoundNet has the architecture as shown in figure 1. Similar to conventional 2D-CNNs used for images, SoundNet has 1D feature maps. The size and number of feature maps for each layer is shown in the figure 1 (b). For example, convolution layer C1 has 16 feature maps each of size 661501 (when input to the network is 30 second long audio samples at 44.1 kHz).

2.2. Identifying redundant feature maps in SoundNet

In previous work [18] on SoundNet, aggregation of feature maps using either sum or max pooling was used to transform them into scalar values. Thus, N feature maps of dimension $1 \times s$ are reduced into an N dimensional feature vector. This is especially useful for dimensionality reduction when feature maps from various layers are being utilized by an ensemble of classifiers. But this approach can utilize all feature maps including the redundant ones.

This can be observed from Figure 2 (A), where (a)-(f) illustrates the feature maps from C3 layer of SoundNet for audio examples of four acoustic scene classes from LITIS Rouen's dataset [19]; shop, hallgare, tubestation and kidgame , (a)-(c) illustrates the feature maps activated differently and (d)-(f) shows the similarly activated feature maps. Figure 2 (B) shows distribution of a given feature map for the same four examples. The first two columns (g), (h) show different distribution for different classes and third column (i) shows similar distribution across classes. With this visual observation, we aim to identify and eliminate the similarly activated feature maps (e.g. 13, 30, 47 as shown in figure 2) for different classes.

3. PROPOSED METHODOLOGY

In this section we describe three statistical frameworks to quantify and eliminate redundant feature maps. We utilise a set of feature maps from a particular layer of SoundNet, which are generated by examples of all classes of interest. Here, the classes come from acoustic scenes. Each example generates N feature maps of size $1 \times s$. Let there be p examples, and let the set of feature maps, standardize to zero mean and unit variance, be denoted $\mathcal{P}$. Thus, $\mathcal{P}$ contains N elements each of size $\mathbb{R}^{p\times s}$. Let each row of the matrix ($\mathbb{R}^{p\times s}$) be denoted as $\mathbf{x}_n \in \mathbb{R}^s$.

ANOVA-based method: In this method, we perform one-way analysis of variance (ANOVA) hypothesis test on each element of size $\mathbb{R}^{p\times s}$ of $\mathcal{P}$ independently. The null hypothesis is that each row $\mathbf{x}_n$ of matrix ($\mathbb{R}^{p\times s}$) have same mean. The hypothesis is validated based on the significance value (*p-value*) given by ANOVA [20]. The procedure to compute *p-value* is explained as follows:

- Compute the "degree of freedom between" (df_b) and "degree of freedom within" (df_w) samples, given as $p-1$ and $(s-1)\times p$ respectively.
- compute the F-value as given in equation 1.

$$F = \frac{(\sum_{n=1}^{p}\sum_{d=1}^{s}(x_{nd}-x_t)^2 - \sum_{n=1}^{p}\sum_{d=1}^{s}(x_{nd}-\bar{x}_n)^2)}{\frac{df_b}{df_w}\times\sum_{n=1}^{p}\sum_{d=1}^{s}(x_{nd}-\bar{x}_n)^2} \tag{1}$$

where $x_{nd} \in \mathbb{R}^{p\times s}$, x_t is the average of all elements in matrix of size $\mathbb{R}^{p\times s}$ (an element of $\mathcal{P}$), $\bar{x}_n$ is the mean of $\mathbf{x}_n$. F denotes the ratio of variance, between and within samples. The p-value can be computed using F-distribution table, given the F, df_w and df_b. If the probability value (p-value) is very high, it means that F-statistic value is small which tells that the samples of p examples for a given feature map have same mean and they are from the same distribution. If the p-value is low, it means that F-value is very large which indicates that the samples are different and they are not from the same distribution, which signifies that the feature map is important.

Entropy-based (DE) method: In this method, the differential entropy (DE) [21] of each element $\in \mathbb{R}^{p\times s}$ of $\mathcal{P}$ indicates the utility of each feature map. A feature map with higher entropy is more important than the feature maps having low entropy value.

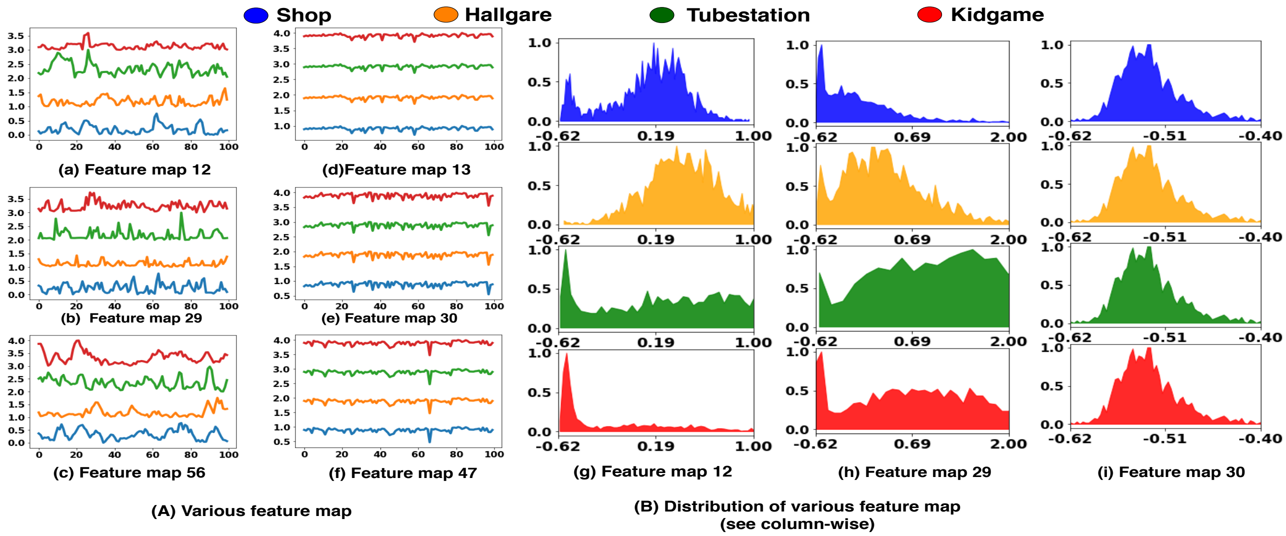
$$\hat{h}_k(\mathbf{x}) = \frac{-1}{p}\sum_{n=1}^{p}\log\hat{p}_k(\mathbf{x_n}), \tag{2}$$

The entropy is given in equation 2, here $\hat{p}_k(\mathbf{x_n})$ is k nearest neighborhood (k-nn) density estimate, $r_k(\mathbf{x_n})$ is the Euclidean distances to the k-nn of $\mathbf{x_n}$ in $\mathcal{P}\backslash\mathbf{x_n}$ and $\pi^{s/2}/\Gamma(s/2+1)$ is the volume of the unit-ball in $\mathbb{R}^s$, $\Gamma(.)$ denotes the gamma function [21].

$$\hat{p}_k(\mathbf{x_n}) = \frac{k}{p-1}\frac{\Gamma(s/2+1)}{\pi^{s/2}}\frac{1}{r_k(\mathbf{x_n})^s} \tag{3}$$

Cosine-similarity (CS) based method: The method measures the cosine angle between a given feature map and other feature maps of an example, denoted as $\cos\phi_{nz}$ where n varies from 1 to p and z varies from 1 to N. Then the standard deviation (σ_z) of the cosine similarity, across the p examples gives an indication of the utility of each feature map and is given by equation 4. A feature map with higher angle deviation is more important than others having low deviation of angle.

$$\sigma_z = \sqrt{\frac{1}{p-1}\sum_{n=1}^{p}(\cos\phi_{nz} - \overline{\cos\phi_z})^2} \tag{4}$$



(a) Feature map 12
(b) Feature map 29
(c) Feature map 56
(d)Feature map 13
(e) Feature map 30
(f) Feature map 47
(g) Feature map 12
(h) Feature map 29
(i) Feature map 30

(A) Various feature map

(B) Distribution of various feature map
(see column-wise)

Fig. 2. (A) Selected feature maps from C3 layer of SoundNet. (a)-(f) shows the feature maps 12, 29 and 56 for examples from four acoustic scene classes: shop, hallgare, tubestation, kidgame (each class in different colours). Feature maps (a)-(c) are more informative, as it shows different responses for different classes. Feature maps (d)-(f) are less important as they respond similarly. (B) Distribution of feature maps 12, 29 and 30 for the same examples. (g) and (h) have different distributions and (i) have similar distributions. (For better visualization in (A), each response for the feature map has been scaled to maximum unit magnitude with unit offset between each consecutive response.)

where $\overline{\cos\phi_z}$ denotes the average of cosine similarity of p examples for a given feature map .

Each of the above methods give the utility of feature maps in terms of probability, informativeness and angle deviation respectively. By ranking the feature maps, their relative importance can be incrementally utilized. The ranking of feature maps in ANOVA-based method is in the ascending order of significance value. While in DE and CS methods, the ranking is in the descending order of entropy and angle deviation respectively. For selection of a set of top ranked feature maps to be used for classification, we propose a greedy technique.

A greedy algorithm for selection of feature maps: Each algorithm in the previous section gives the N feature maps ranked (Λ) in order of their utility. Our objective is now to choose the top l feature maps so that the remaining $N-l$ feature maps can be ignored. A heuristic that can be used here is the representation ability of the top m feature maps, as m varies from 1 to l. This is done by estimating the Kullback-Leibler divergence (D_{KL}) of the discrete probability distribution (h) of the top m feature maps with respect to the discrete probability distribution (H) of all the N feature maps and is denoted as $\Psi[m] = D_{\mathrm{KL}}(\mathrm{h}||\mathrm{H})$.

The stopping criteria for the greedy procedure is when the cumulative absolute difference of KL-divergence changes by less than ϵ as additional ρ feature maps are added. The greedy algorithm is summarized in algorithm 1.

Algorithm 1 A greedy algorithm to select top-l feature maps

```
input : Λ: Ranked indexes of feature maps.
        H: Distribution of all feature maps of $p$ examples.
        $\mathcal{P}$: A set of $N$ elements each of size $\mathbb{R}^{p\times s}$.
        $\rho$ and $\epsilon$ are stopping criterion hyper-parameters.
output: Indexes of top-$l$ feature maps ($\xi$).
# Compute KL div. b/w partial and full set of feature maps.
for $m \leftarrow 1$ to size($\Lambda$) do
    $\lambda = [\,]$                    # initialize partial set as an empty set.
    $\lambda$.append($\Lambda[1:m]$)    # append first $m$ feature map index.
    h : distribution of feature maps indexed by $\lambda$ of $p$ examples.
    $\Psi[m] = D_{\mathrm{KL}}(\mathrm{h}||\mathrm{H})$
end
# Select top l feature maps meeting stopping criterion.
for $l \leftarrow 1$ to size($\Psi$) do
    # Calculate the change in KL div. b/w each consecutive $\rho$ indexes.
    for $w \leftarrow 0$ to $\rho - 1$ do
        $\delta[w] = \Psi(l+w) - \Psi(l+w+1)$
    end
    # Verify the stopping criterion.
    if $\|\delta\|_1 \le \epsilon$ then
        $\xi = \Delta[1:l]$
        break                           # return $\xi$, the selected indexes.
    end if
end
```

4. PERFORMANCE EVALUATION

4.1. Datasets Used and Experimental setup

We use two audio scene classification (ASC) datasets for evaluation purposes: (a) TUT DCASE 2016 ASC dataset [22], comprising of a development set and an evaluation set, each of which has 15 audio scene classes, and, (b) Environmental Sound Classification (ESC-50) dataset comprising of 50 audio scene classes [23]. An ensemble based classification procedure as proposed in [18] is opted for classification. The set $\mathcal{P}$ used to determine which feature maps to prune is a subset of the DCASE development data. This consists of 10 examples from each of the 15 classes, resulting in $p = 150$. This subset of the data is not utilised for evaluating the proposed framework.

The pruning procedure described in section 3 is applied to each of the 15 layers of SoundNet (from P1 to C7 including the batch normalization layer as shown in figure 1). The number of nearest neighbors (k) in entropy-based method (section 3) is set equal to 10. The stopping criteria is met by setting ϵ and ρ, 0.01 and 10 respectively. The performance of the proposed framework is measured in terms of classification accuracy over 4-fold and 5-fold cross-validation for DCASE and ESC datasets respectively. The overall proposed framework is shown in figure 3.

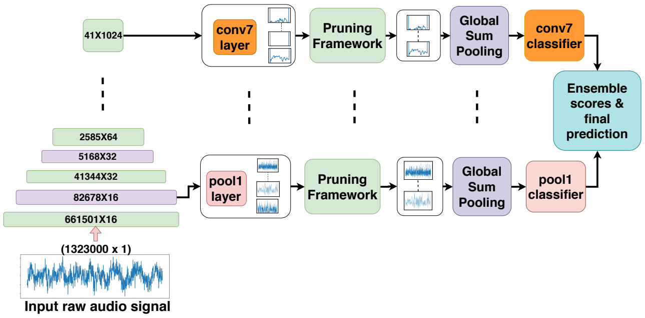


Fig. 3. Evaluation framework with ensemble of classifiers after sum pooling in each layer. Proposed pruning is applied to select important feature maps from each layer.

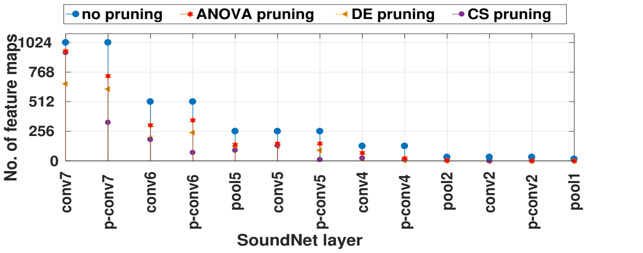


Fig. 4. Layer-wise number of pruned feature maps for different pruning methods.

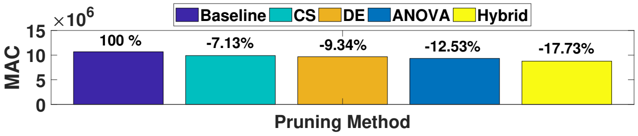


Fig. 5. Multiplication and accumalation (MAC) operations to compute feature representation for DCASE 2016 ASC dataset with different pruning methods. The quantity over each bar shows percentage (%) of saved operations with respect to baseline.

4.2. Results and Analysis

Figure 4 shows the number of redundant feature maps obtained after the three pruning procedures. This figure indicates that majority of the feature maps in the later layers (conv7 and p-conv7) are redundant and can be removed. The number of redundant feature maps reduces towards the earlier layers (approximately all the feature maps are not redundant for conv2 and p-conv2 layers). The number of feature maps retained after ANOVA, DE and CS pruning is approx. 32%, 50% and 57% respectively (averaged across all layers).

A *hybrid* method, which uses the intersection of the redundant feature maps returned by each method is also evaluated. The hybrid method retains about 11% from all feature maps of SoundNet.

Figure 6 shows how the KL divergence varies as more and more feature maps are included in the representation. In the figure, the stopping criteria of $\epsilon = 0.01$ is met when feature maps with rank 105-115 are appended to the set of selected feature maps.

Computational complexity for feature representation: The selected feature maps are further collapsed to l scalar values by using sum pooling before feeding it to the classifier ensemble. The total multiply and accumulation (MAC) operations required to compute the scalar values from an s-length feature map is of twice the order of s (s multiplications and s-1 additions). We define (R_{CC}) as total reduction in computational complexity (CC) for feature representation of all layers and it is given as follows:

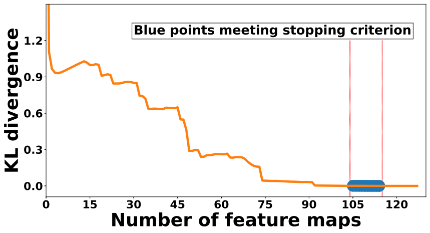


Fig. 6. Reduction in KL-divergence as more feature maps are selected in conv4 layer using CS framework.

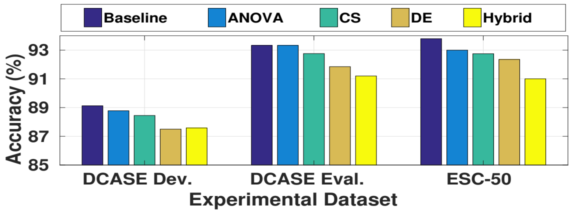


Fig. 7. Performance with different pruning methods.

$$R_{\text{CC}} = \frac{\text{CC}_{\text{baseline}} - \text{CC}_{\text{reduced}}}{\text{CC}_{\text{baseline}}} * 100$$

Here, $\text{CC}_{\text{baseline}}$ represents the total MAC for feature representation in the baseline network and $\text{CC}_{\text{reduced}}$ is the total MAC after eliminating redundant feature maps. The R_{CC} for ANOVA, DE, CS and hybrid methods is given in figure 5. The R_{CC} for hybrid method is approx. 18%, since it chooses smallest subset of feature maps as compared to the other methods.

Dimensionality reduction and ensemble classifier complexity: The complexity required to train a non-linear support vector machine (SVM) for each layer is $O(N \times M)$; N is the number of features maps and M is total number of examples. Since for each layer the reduction in dimensionality is equivalent to the number of pruned feature maps for the particular layer as shown in figure 4, the model complexity for each layer reduces by $O((N - l) \times M)$; $N - l$ is the number of pruned feature map for the particular layer.

Performance variation: The accuracy of the baseline network and the pruned network (with different pruning frameworks) for the two different datasets is shown in figure 7. For all pruning methods, including the hybrid method, the reduction in accuracy is not more than 0.8% and 1.63% for DCASE development and evaluation datasets respectively. The same set of feature maps derived from the set $\mathcal{P}$ is used on the ESC-50 dataset. The performance on the ESC dataset, which has 50 scene classes, does not degrade beyond 2.8%. This shows the generalization ability of the proposed framework. We note that ANOVA-based method selects the smallest subset of important feature maps with least reduction in performance as compared to DE and CS-based methods.

5. CONCLUSION

In this paper, we propose a statistical pruning framework to eliminate redundant feature maps learned at various hidden layers in pre-trained audio network. The proposed framework reduces computational complexity and dimensionality reduction for layer-wise analysis while using an ensemble of classifiers. In addition to this, the proposed framework preserves the baseline architecture fully and does not require fine-tuning.

6. REFERENCES

[1] Andre Esteva, Brett Kuprel, Roberto A. Novoa, Justin Ko, Susan M. Swetter, Helen M. Blau, and Sebastian Thrun, “Dermatologist-level classification of skin cancer with deep neural networks,” *Nature*, vol. 542, no. 7639, pp. 115, 2017.

[2] Jort F. Gemmeke, Daniel P. W. Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R. Channing Moore, Manoj Plakal, and Marvin Ritter, “Audio set: An ontology and human-labeled dataset for audio events,” in *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*. IEEE, 2017, pp. 776–780.

[3] Yusuf Aytar, Carl Vondrick, and Antonio Torralba, “Soundnet: Learning sound representations from unlabeled video,” in *Advances in Neural Information Processing Systems*, 2016, pp. 892–900.

[4] Shawn Hershey, Sourish Chaudhuri, Daniel P. W. Ellis, Jort F. Gemmeke, Aren Jansen, R. Channing Moore, Manoj Plakal, Devin Platt, Rif A. Saurous, Bryan Seybold, et al., “CNN architectures for large-scale audio classification,” in *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*. IEEE, 2017, pp. 131–135.

[5] Loris Nanni, Yandre M. G. Costa, Diego Rafael Lucio, Carlos N. Silla Jr, and Sheryl Brahnam, “Combining visual and acoustic features for audio classification tasks,” *Pattern Recognition Letters*, vol. 88, pp. 49–56, 2017.

[6] Tawfiq Salem, Menghua Zhai, Scott Workman, and Nathan Jacobs, “A multimodal approach to mapping soundscapes,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 2524–2527.

[7] Michele Merler, Dhiraj Joshi, Khoi-Nguyen C. Mac, Quoc-Bao Nguyen, Stephen Hammer, John Kent, Jinjun Xiong, Minh N. Do, John R. Smith, and Rogerio S. Feris, “The excitement of sports: Automatic highlights using audio/visual cues,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 2520–2523.

[8] Tien-Ju Yang, Yu-Hsin Chen, and Vivienne Sze, “Designing energy-efficient convolutional neural networks using energy-aware pruning,” *arXiv preprint arXiv:1611.05128*, 2016.

[9] Ruichi Yu, Ang Li, Chun-Fu Chen, Jui-Hsin Lai, Vlad I. Morariu, Xintong Han, Mingfei Gao, Ching-Yung Lin, and Larry S. Davis, “Nisp: Pruning networks using neuron importance score propagation,” *Preprint at https://arxiv.org/abs/1711.05908*, 2017.

[10] Song Han, Huizi Mao, and William J. Dally, “Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding,” *arXiv preprint arXiv:1510.00149*, 2015.

[11] Ji Lin, Yongming Rao, Jiwen Lu, and Jie Zhou, “Runtime neural pruning,” in *Advances in Neural Information Processing Systems*, 2017, pp. 2181–2191.

[12] Sajid Anwar, Kyuyeon Hwang, and Wonyong Sung, “Structured pruning of deep convolutional neural networks,” *ACM Journal on Emerging Technologies in Computing Systems (JETC)*, vol. 13, no. 3, pp. 32, 2017.

[13] Hao Li, Asim Kadav, Igor Durdanovic, Hanan Samet, and Hans Peter Graf, “Pruning filters for efficient convnets,” *arXiv preprint arXiv:1608.08710*, 2016.

[14] Jian-Hao Luo, Jianxin Wu, and Weiyao Lin, “Thinet: A filter level pruning method for deep neural network compression,” *arXiv preprint arXiv:1707.06342*, 2017.

[15] Adam Polyak and Lior Wolf, “Channel-level acceleration of deep face representations,” *IEEE Access*, vol. 3, pp. 2163–2175, 2015.

[16] Tao Kong, Anbang Yao, Yurong Chen, and Fuchun Sun, “Hypernet: Towards accurate region proposal generation and joint object detection,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 845–853.

[17] Sean Bell, C. Lawrence Zitnick, Kavita Bala, and Ross Girshick, “Inside-outside net: Detecting objects in context with skip pooling and recurrent neural networks,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2874–2883.

[18] Arshdeep Singh, Anshul Thakur, Padmanabhan Rajan, and Arnav Bhavsar, “A layer-wise score level ensemble framework for acoustic scene classification,” in *2018 26th European Signal Processing Conference (EUSIPCO)*, 2018, pp. 842–846.

[19] Alain Rakotomamonjy and Gilles Gasso, “Histogram of gradients of time-frequency representations for audio scene classification,” *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, vol. 23, no. 1, pp. 142–153, 2015.

[20] W. Penny and R. Henson, “Analysis of variance,” *Statistical parametric mapping: The analysis of functional brain images*, pp. 166–177, 2006.

[21] Fernando Pérez-Cruz, “Estimation of information theoretic measures for continuous random variables,” in *Advances in neural information processing systems*, 2009, pp. 1257–1264.

[22] Annamaria Mesaros, Toni Heittola, and Tuomas Virtanen, “TUT database for acoustic scene classification and sound event detection,” in *Signal Processing Conference (EUSIPCO), 2016 24th European*. IEEE, 2016, pp. 1128–1132.

[23] Karol J Piczak, “ESC: Dataset for environmental sound classification,” in *Proceedings of the 23rd ACM international conference on Multimedia*. ACM, 2015, pp. 1015–1018.