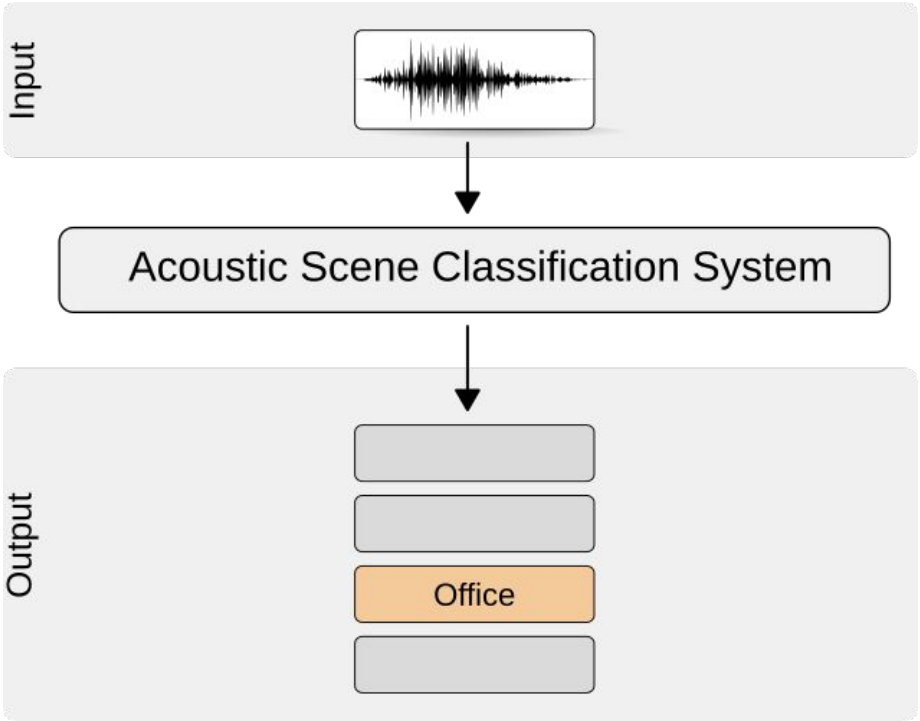
IEEE DCASE 2017
Task 1: Acoustic Scene Classification
Using
Shift-Invariant Kernels and Random Features

Abelino Jimenez, Benjamin Elizalde, Bhiksha Raj
Carnegie Mellon University

Acoustic Scene Classification



Examples of acoustic scenes

- Bus - traveling by bus in the city (vehicle)
- Cafe / Restaurant - small cafe/restaurant (indoor)
- Car - driving or traveling as a passenger, in the city (vehicle)
- City center (outdoor)
- Forest path (outdoor)
- Grocery store - medium size grocery store (indoor)
- Home (indoor)
- Lakeside beach (outdoor)
- Library (indoor)
- Metro station (indoor)
- Office - multiple persons, typical work day (indoor)
- Residential area (outdoor)
- Train (traveling, vehicle)
- Tram (traveling, vehicle)
- Urban park (outdoor)

Outline

Introduction and Method

Experiments and Results

Ongoing work

Some state of the art approaches combine large-dimensional features and SVMs

(Geiger, 2013) *Large-scale audio feature extraction and SVM for acoustic scene classification.* 6,553 dim

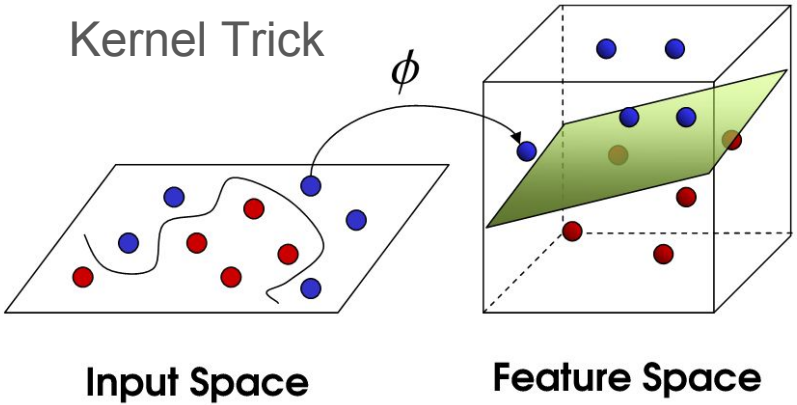
(Metze, 2014) *Improved Audio Features for Large-scale Multimedia Event Detection.* 4,096 dim

(Rakotomamonjy,2016) *Enriched Supervised Feature Learning for Acoustic Scene Classification.* 2,000 dim

(Zhang, 2017) *Learning Audio Sequence Representations for Acoustic Event Classification.* 6,373 dim

(Arandjelovic, 2017) *Look, Listen and Learn.* 6,144 dim

SVMs may employ nonlinear functions, but the computation complexity increases



$$\mathcal{K}(\mathbf{x}_i, \mathbf{x}_j) = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle \quad \forall i, j$$

Kernel matrix complexity:

Training $O(k^2n)$, Testing $O(kn)$

k= number of samples

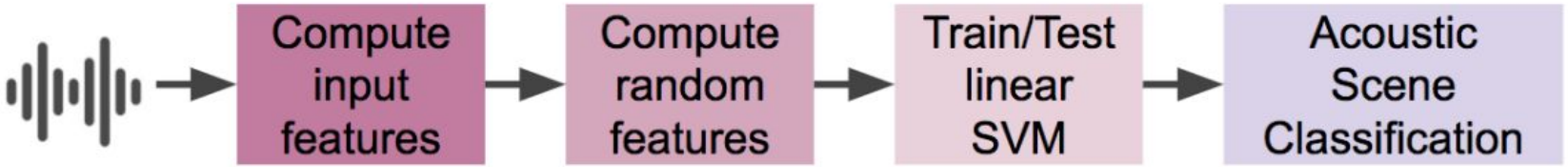
n= dimensionality of features

$$\phi : \mathbb{R}^n \rightarrow \mathbb{R}^q \quad (n << q)$$

Random Features

Consist of mapping the input/original features to a randomized lower-dimensional feature space.

Then, the RFs are passed to a linear SVM to approximate a nonlinear SVM.



(Rahimi, 2008) “Random features for large-scale kernel machines”

Function to compute Random Features

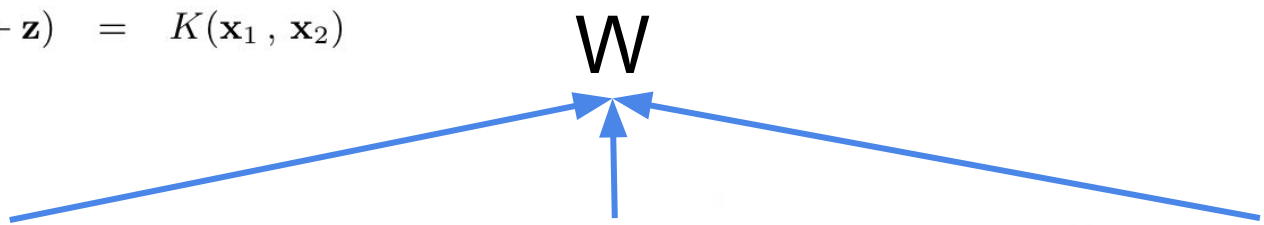
$$\mathcal{K}(\mathbf{x}_i, \mathbf{x}_j) = \langle \phi(\mathbf{x}_i)\,,\, \phi(\mathbf{x}_j) \rangle \approx \langle \Phi_{RF}(\mathbf{x}_i), \Phi_{RF}(\mathbf{x}_j) \rangle \quad \forall i,\, j$$

$$\Phi_{RF}(\mathbf{x}) = \sqrt{\frac{2}{M}} \cos\left(\mathbf{W}\mathbf{x} + \mathbf{b}\right)$$

- Input features
- Fixed value
- Kernel dependent **W** : MxN
- Uniform distribution **b** : Mx1

Random Features for shift-invariant kernels

$$K(\mathbf{x}_1 + \mathbf{z}\,,\, \mathbf{x}_2 + \mathbf{z}) \quad = \quad K(\mathbf{x}_1\,,\, \mathbf{x}_2)$$



Gaussian

$$w_{ij} \sim \mathcal{N}(0, 2\gamma)$$

$$K(\mathbf{x}_1, \mathbf{x}_2) \quad = \quad \exp\left(-\gamma \|\mathbf{x}_1 - \mathbf{x}_2\|_2^2\right)$$

Laplace

$$w_{ij} \sim \text{Cauchy}(0, \gamma)$$

$$K(\mathbf{x}_1, \mathbf{x}_2) \quad = \quad \exp\left(-\gamma \|\mathbf{x}_1 - \mathbf{x}_2\|_1\right)$$

Cauchy

$$w_{ij} \sim \text{Laplace}(0, \gamma)$$

$$K(\mathbf{x}_1, \mathbf{x}_2) \quad = \quad \prod_{i=1}^{N} \frac{1}{1 + \gamma^2 (x_{1i} - x_{2i})^2}$$

SVM prediction with Random Features

$$f(\mathbf{x}_{\text{test}}) = \sum_i \alpha_i y_i \mathcal{K}(\mathbf{x}_i, \mathbf{x}_{\text{test}}) + \delta \qquad \text{keep } \{\alpha_i, y_i, \mathbf{x}_i, \delta\}$$

$$f(\mathbf{x}_{\text{test}}) = \omega^\top \Phi_{RF}(\mathbf{x}_{\text{test}}) + \delta \qquad \text{keep } \{\omega,\ \text{seed}\ , \delta\}$$

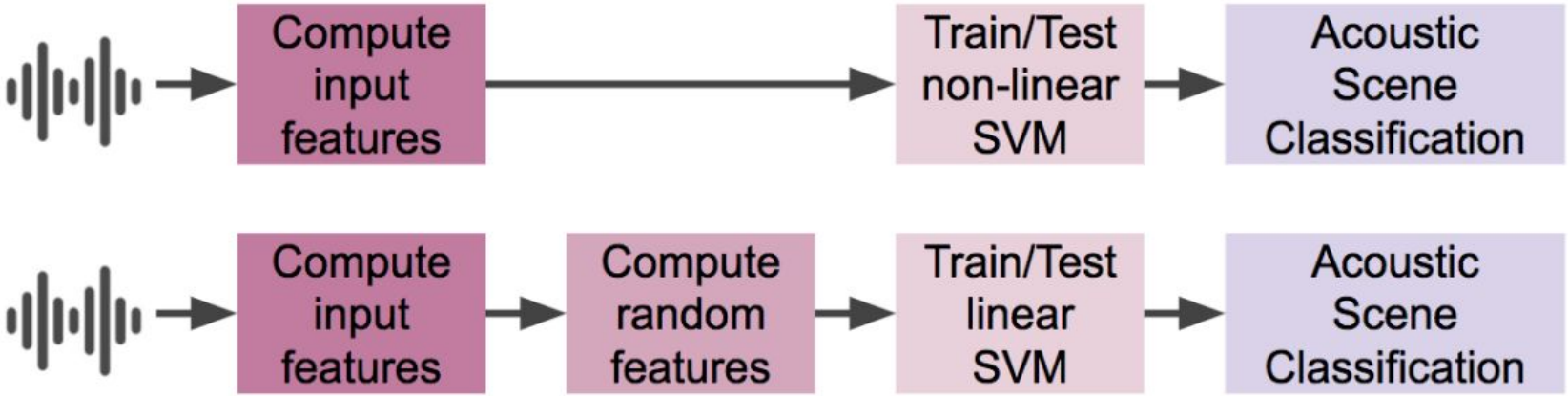
Outline

Introduction and Method

Experiments and Results

Ongoing work

Compare performance with/without Random Features



Experimental setup

Audio: 3-5 minutes duration from 15 scenes (e.g. bus, park, library)

Input features: 6,553 dims, cepstral, spectral, energy related, voicing, functionals (Geiger, 2013)

Random Features: approximate Gaussian, Laplacian, Cauchy

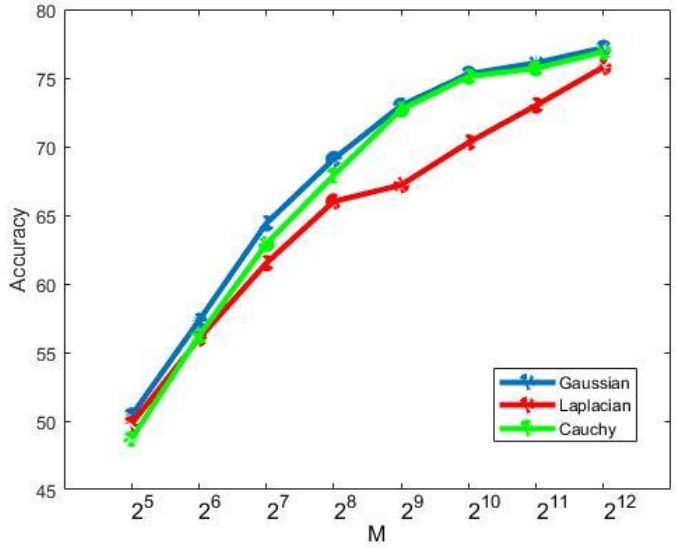
Classifier: SVM with Gaussian, Laplacian, Cauchy

Metric: Accuracy

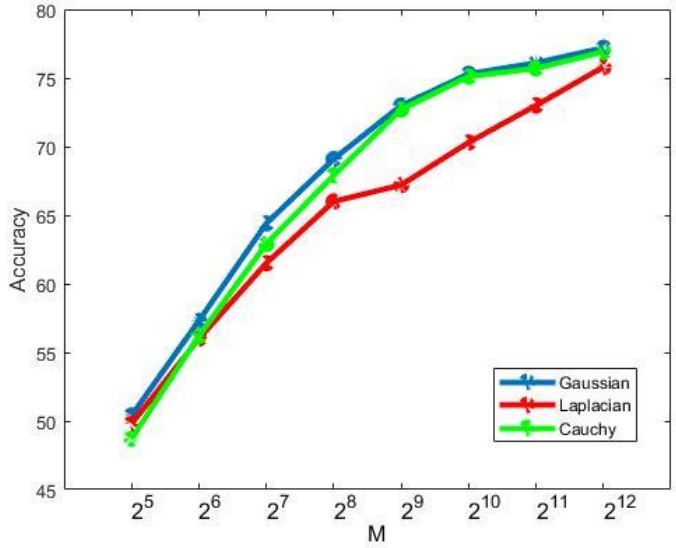
Nonlinear SVM outperformed MLP baseline

Acoustic scene	Baseline	Gaussian Kernel	Laplacian Kernel	Cauchy Kernel
Beach	75.3 %	78.8 %	77.2 %	77.9 %
Bus	71.8 %	93.6 %	92.0 %	92.3 %
Cafe/Restaurant	57.7 %	76.9 %	82.7 %	78.5 %
Car	97.1 %	94.9 %	94.2 %	95.5 %
City center	90.7 %	91.0 %	92.3 %	89.4 %
Forest path	79.5 %	89.1 %	85.9 %	87.2 %
Grocery store	58.7 %	74.7 %	74.7 %	74.0 %
Home	68.6 %	66.3 %	67.3 %	66.3 %
Library	57.1 %	65.7 %	58.3 %	65.1 %
Metro station	91.7 %	82.7 %	83.7 %	83.3 %
Office	99.7 %	89.7 %	92.9 %	90.4 %
Park	70.2 %	65.1 %	61.5 %	60.9 %
Residential area	64.1 %	65.7 %	68.3 %	63.5 %
Train	58.0 %	57.7 %	65.7 %	61.9 %
Tram	81.7 %	82.7 %	84.3 %	81.7 %
Overall	**74.8 %**	**78.3 %**	**78.8 %**	**77.9 %**

For RFs, as M increases, performance approximates nonlinear kernels

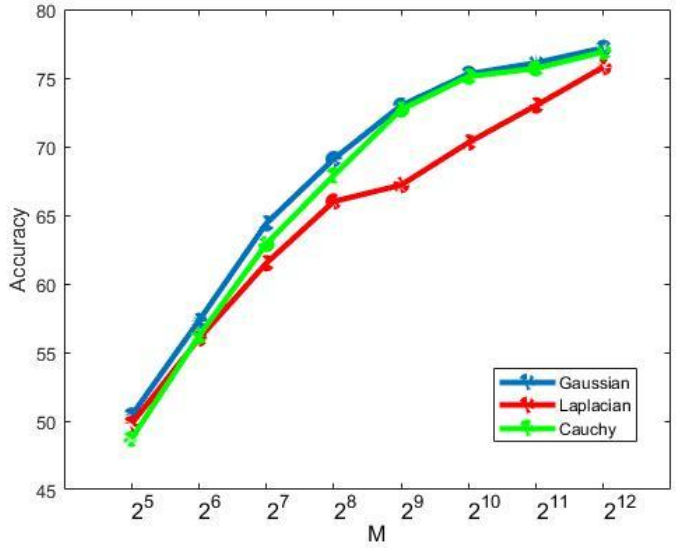


For RFs, as M increases, performance approximates nonlinear kernels



Dimensionality	Gaussian Kernel	Laplacian Kernel	Cauchy Kernel
6,553; $>2^{12}$	**78.3 %**	**78.8 %**	**77.9 %**

For RFs, as M increases, performance approximates nonlinear kernels



Dimensionality	Gaussian Kernel	Laplacian Kernel	Cauchy Kernel
6,553; $>2^{12}$	**78.3 %**	**78.8 %**	**77.9 %**
4,096; 2^{12}	77.2 %	75.8 %	76.9 %

37%

Outline

Introduction and Method

Experiments and Results

Ongoing work

Discretize random features (Hashing) to change real-valued vectors into bits

$$K(\mathbf{x}, \mathbf{y}) = \langle \phi(\mathbf{x}), \phi(\mathbf{y}) \rangle \approx \langle H(\mathbf{x}), H(\mathbf{y}) \rangle$$

Hashing Trick

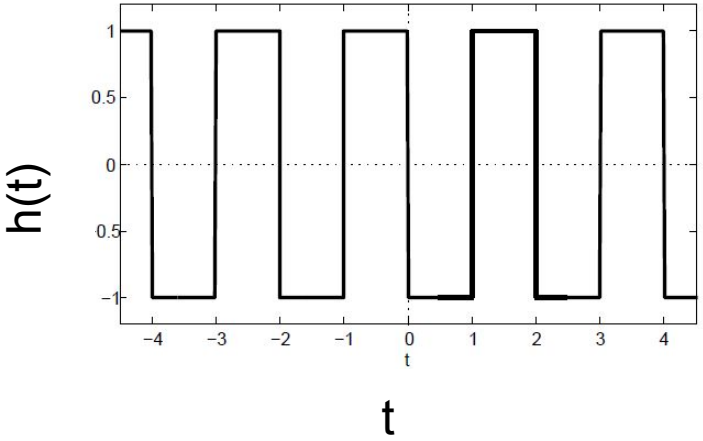
Abelino Jimenez, Benjamin Elizalde, Bhiksha Raj, *"Acoustic Scene Classification Using Discrete Random Hashing for Laplacian Kernel Machines",* in submission to ICASSP 2018

Discretize random features (Hashing) to change real-valued vectors into bits

$$K(\mathbf{x},\mathbf{y}) = \langle \phi(\mathbf{x}), \phi(\mathbf{y}) \rangle \approx \langle H(\mathbf{x}), H(\mathbf{y}) \rangle$$

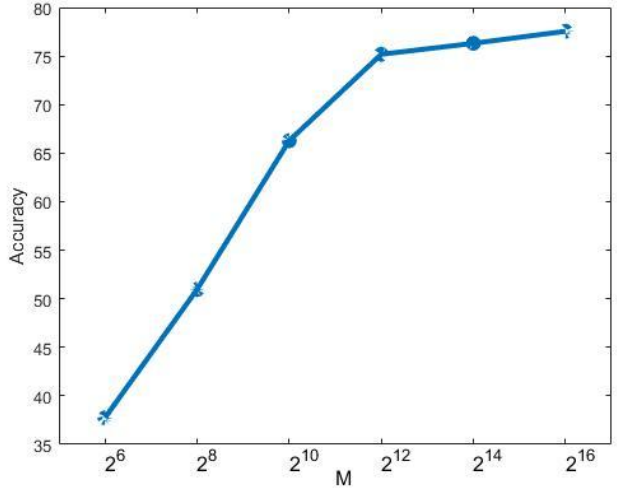
Hashing Trick

$$H_{A,U}(\mathbf{x}) = \frac{1}{\sqrt{M}} h(A\mathbf{x} + U)$$



Abelino Jimenez, Benjamin Elizalde, Bhiksha Raj, *"Acoustic Scene Classification Using Discrete Random Hashing for Laplacian Kernel Machines",* in submission to ICASSP 2018

Reduces representation up to six orders of magnitude



Method	Accuracy	# of Bits
DCASE Challenge	74.8%	-
Laplacian Kernel	78.6%	$> 2^{18}$
Random features $M = 2^{12}$	75.8%	2^{18}
Hashing $M = 2^{12}$	75.2%	2^{12}

Summary

Random features with linear SVM approximates well nonlinear SVM.

RFs reduced dimensionality by 37% with minimal loss of performance.

Hashing can also reduce storage up to 6 orders of magnitude.

Allows bit-based operations (XOR for similarity)

Speeds up transmissions and processing (self-training, boosting)