Time-Frequency Image Resizing Using Interpolation for Acoustic Event Recognition with Convolutional Neural Networks

Roneel V. Sharan
School of Information Technology and Electrical Engineering
The University of Queensland
Brisbane, Australia
r.sharan@uq.edu.au

Tom J. Moir
School of Engineering, Computer and Mathematical Sciences
Auckland University of Technology
Auckland, New Zealand
tom.moir@aut.ac.nz

***Abstract*—Convolutional neural networks (CNN) are being increasingly used for audio signal classification applications, including acoustic event recognition. CNN is an image classifier and acoustic event signals are often represented using time-frequency image for this purpose. However, the length or duration of the sound event signals can vary greatly and an important consideration is how to equally size time-frequency images for classification using CNN. In this paper, we use techniques from digital image processing to address this problem. In particular, we apply interpolation-based image resizing techniques to form equally sized time-frequency representations. We consider nearest-neighbor, bilinear, bicubic, and Lanczos kernel interpolation methods for this purpose. A database containing 50 sound event classes with sound events of varying duration is used to evaluate the classification performance of these resized time-frequency images. The results show that the time-frequency images resized using bicubic and Lanczos kernel interpolation methods give a much improved classification performance than the conventional time-frequency image representation.**

Keywords—acoustic event recognition, convolutional neural network, image resize, interpolation, spectrogram, time-frequency image

I. Introduction

Convolutional neural networks (CNN) became popular with their superior classification performance in image classification tasks, such as on the ImageNet dataset [1]. It wasn't long before CNN was adapted for audio classification tasks, such as speech [2] and acoustic event classification [3], producing improved results against various baseline methods.

CNN is an image classifier. Time-frequency image representation of audio signals is the most common approach for audio signal classification using CNN, as seen in [2, 3] and various other literature. However, unlike image classification tasks where input images are often of the same dimension, the signal length of acoustic events can vary greatly. As such, an important consideration is how to represent acoustic events of different signal lengths to same time-frequency image dimensions as required for classification with CNN.

A simple approach is to divide the signal into equal number of frames and then compute the fast Fourier transform (FFT) of all frames with a fixed number of FFT points. However, this method has a disadvantage that having a large number of FFT points would results in a large time-frequency image representation which would increase the computational costs with CNN classification. On the other hand, having a small number of FFT points results in a large spacing between frequency components and, as such, distinguishing frequency characteristics may not be captured well enough resulting in a poor classification performance.

A compromise for this problem is to compute FFT using a large number of FFT points and then use filters to reduce the number of frequency components using subband energies. The filter reduces the number of frequency components by computing the filter bank energies thereby retaining the frequency characteristics to some extent. Two commonly used filters for this purpose are moving average filter and mel-filter. In addition, gammatone filters, the frequency components of which are based on the human auditory system, have been used in recent work [3]. The corresponding time-frequency image representations are referred as smoothed spectrogram, mel-spectrogram, and cochleagram or gammatone-spectrogram, respectively [3-5].

Furthermore, the duration of acoustic event signals can vary greatly and capturing distinguishing frequency characteristics with small frame lengths could be difficult. In this case, it is possible to divide audio signals into fixed frame length, instead of fixed number of frames, which means the number of frames for each acoustic event signal will vary, dependent on the length of the sound signal. Thereafter, image resizing techniques can be applied to the spectrogram image to ensure all CNN input image are of the same dimension [6].

In this work, we focus on the less conventional approach of time-frequency image resizing inspired from work in the field of digital image processing. In particular, we look at image interpolation methods for this purpose. Five common interpolation techniques are studied. These are nearest-neighbor interpolation, bilinear interpolation, bicubic interpolation, Lanczos-2 kernel interpolation, and Lanczos-3 kernel interpolation.

The organization of the rest of the paper is as follows. In Section II we give details of the different time-frequency image resizing techniques. Experimental setup and results are presented in Section III and conclusions in Section IV.

II. Time-Frequency Image Formation

An overview of conventional time-frequency image formation, frequency-filtered time-frequency image formation, and resized time-frequency image formation is given in Fig. 1. The use of filters in time-frequency image formation, such as smoothed spectrogram using moving

978-1-7281-2177-2/19/$31.00 ©2019 IEEE

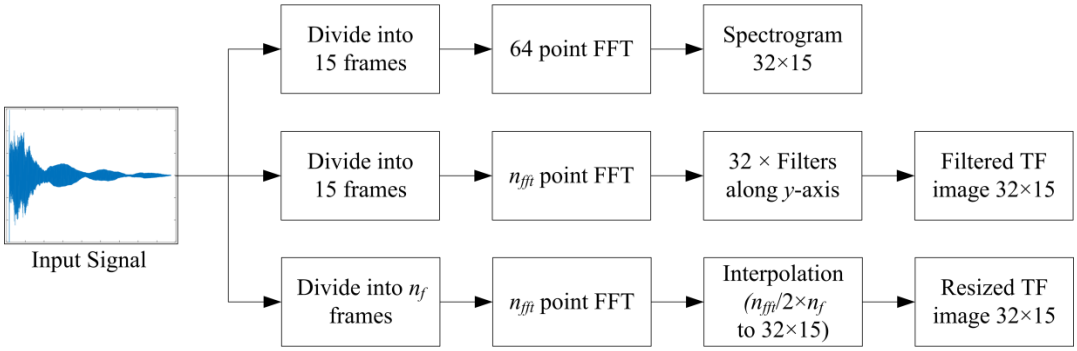


Fig. 1. An overview of the different time-frequency image formation techniques.

average filter, mel-spectrogram using mel-filter, and cochleagram using gammatone filter, has been discussed in [3]. In this work, we focus on time-frequency image formation using image resizing techniques. The CNN input layer image dimension is chosen as 32×15, same as [3]. Similar approach has also been used in speech classification [2].

A. Spectrogram

In forming the spectrogram image, FFT is applied to the windowed signal as

$$X(k,t)=\sum_{n=0}^{N-1} x(n)w(n)e^{\frac{-2\pi ikn}{N}}, \qquad k=0,...,N-1 \quad (1)$$

where N represents the window length, $x(n)$ represents the signal in time-domain, $X(k,t)$ is the k^{th} harmonic frequency ($f(k)=kF_s/N$) for the t^{th} frame, the sampling frequency is given by F_s, and the window function by $w(n)$.

The log of the magnitude of the FFT values are computed to form the spectrogram as

$$S(k,t)=\log\left(\left|X(k,t)\right|\right). \quad (2)$$

In forming a 32×15 dimensional spectrogram image, each signal is divided into 15 frames and the overlap between frames is set to 50%. The frames are windowed using a Hamming window and 64 points are used in computing the FFT values. This results in a 32×15 dimensional spectrogram image. Illustration of a spectrogram image of a sample sound event signal is shown in Fig. 2. The jet colorspace is used for mapping the grayscale spectrogram for better visualization.

All time-frequency representations illustrated in this paper utilize the same sound event signal and the frequency range in each case is same, from 0Hz to the Nyquist frequency of 22,050Hz.

B. Resizing Using Interpolation

Interpolation is a commonly used technique in digital image processing to scale or resize images. It is a process through which a continuous image can be spatially defined from discrete samples. A common method of interpolation is convolving an image with a small kernel with weight

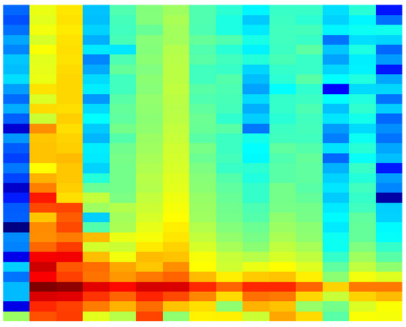


Fig. 2. Spectrogram image of size 32×15 using 15 frames, 50% overlap between frames, and 64 FFT points.

coefficients. Various kernels exist for this purpose. Some popular kernels for interpolation by convolution are nearest-neighbor, bilinear, bicubic, Lanczos-2, and Lanczos-3.

1) Nearest-Neighbor Interpolation

Nearest neighbor interpolation is a very simple interpolation algorithm which selects the value of the closest or nearest neighboring point [7], that is,

$$R_{NN}(x,y)=S_{[x][y]} \quad (3)$$

where [.] denotes rounding to the nearest integer.

The nearest-neighbor interpolation kernel in one dimension can be given as

$$k(x)=\begin{cases}1, & |x|<0.5\\ 0, & otherwise.\end{cases} \quad (4)$$

2) Bilinear Interpolation

Bilinear interpolation can be seen as an extension of linear interpolation in both x and y directions. The solution to the interpolation can be given as

$$R_{BL}(x,y)=a_0+a_1x+a_2y+a_3xy \quad (5)$$

where a_0, a_1, a_2 and a_3 are determined from the four nearest neighbors of (x,y). It can be implemented using a triangular kernel as

$$k(x)=\begin{cases}1-|x|, & |x|<1\\ 0, & \textit{otherwise.}\end{cases} \tag{6}$$

3) Bicubic Interpolation

Unlike bilinear, bicubic resamples using 16 neighboring pixels and the interpolation can be given as

$$R_{BC}(x,y)=\sum_{i=0}^{3}\sum_{j=0}^{3}a_{ij}x^{i}y^{j} \tag{7}$$

where a_{ij} are determined from the sixteen nearest neighbors of (x,y). Bicubic interpolation can be achieved by applying convolution with the following kernel [8]

$$k(x)=\begin{cases}\frac{3}{2}|x|^3-\frac{5}{2}|x|^2+1, & |x|\le 1\\ -\frac{1}{2}|x|^3+\frac{5}{2}|x|^2-4|x|+2, & 1<|x|\le 2\\ 0, & \textit{otherwise.}\end{cases} \tag{8}$$

4) Lanczos Kernel Interpolation

The Lanczos kernel is a normalized sinc function [9] windowed by the sinc window which can be equivalently written as [10]

$$L(x)=\begin{cases}1, & x=0\\ \frac{a\sin(\pi x)\sin(\pi x/a)}{\pi^2x^2}, & -a\le x<a \text{ and } x\ne 0\\ 0, & \textit{otherwise}\end{cases} \tag{9}$$

where a is a positive integer; the kernel is referred as Lanczos-2 when $a=2$ and Lanczos-3 when $a=3$.

The Lanczos interpolation can then be computed as [7]

$$R_L(x)=\sum_{i=\lfloor x\rfloor-a+1}^{\lfloor x\rfloor+a}S_iL(x-i) \tag{10}$$

where $\lfloor x\rfloor$ is the floor function of x, a is the filter size, and S_i is a one dimensional signal. The Lanczos kernel in two dimensions can be given as $L(x,y)=L(x)L(y)$.

Illustration of resized spectrograms, from the original spectrogram representation of Fig. 3(a), using nearest-neighbor interpolation, bilinear interpolation, bicubic interpolation, Lanczos-2 kernel interpolation, and Lanczos-3 kernel interpolation are given in Fig. 3(b)-(f), respectively. The log spectrogram image is resized in each case.

III. Experimental Evaluation

A. Dataset

The Real World Computing Partnership (RWCP) Sound Scene Database (SSD) in Real Acoustical Environments [11] is utilized in this work. The final dataset has a total of 4000 manually segmented sound event files, 80 files per each of the 50 classes. The sampling frequency of the signals is 44,100 Hz with 16-bit resolution.

B. Experimental Setup

The network architecture and parameter settings of the CNN used in this work is same as [3]. The classifier is trained and validated with 50 samples per class for each spectrogram representation. The remaining 30 samples are used for testing the trained model. Therefore, the classifier is trained and validated using 2500 samples and the trained model is tested on the remaining 1500 samples. The classification accuracy, defined as the percentage of *correctly classified test samples* divided by the *total number of test samples*, is used to evaluate the performance of each time-frequency representation. The classification accuracy in each case was averaged over 10 runs.

C. Results

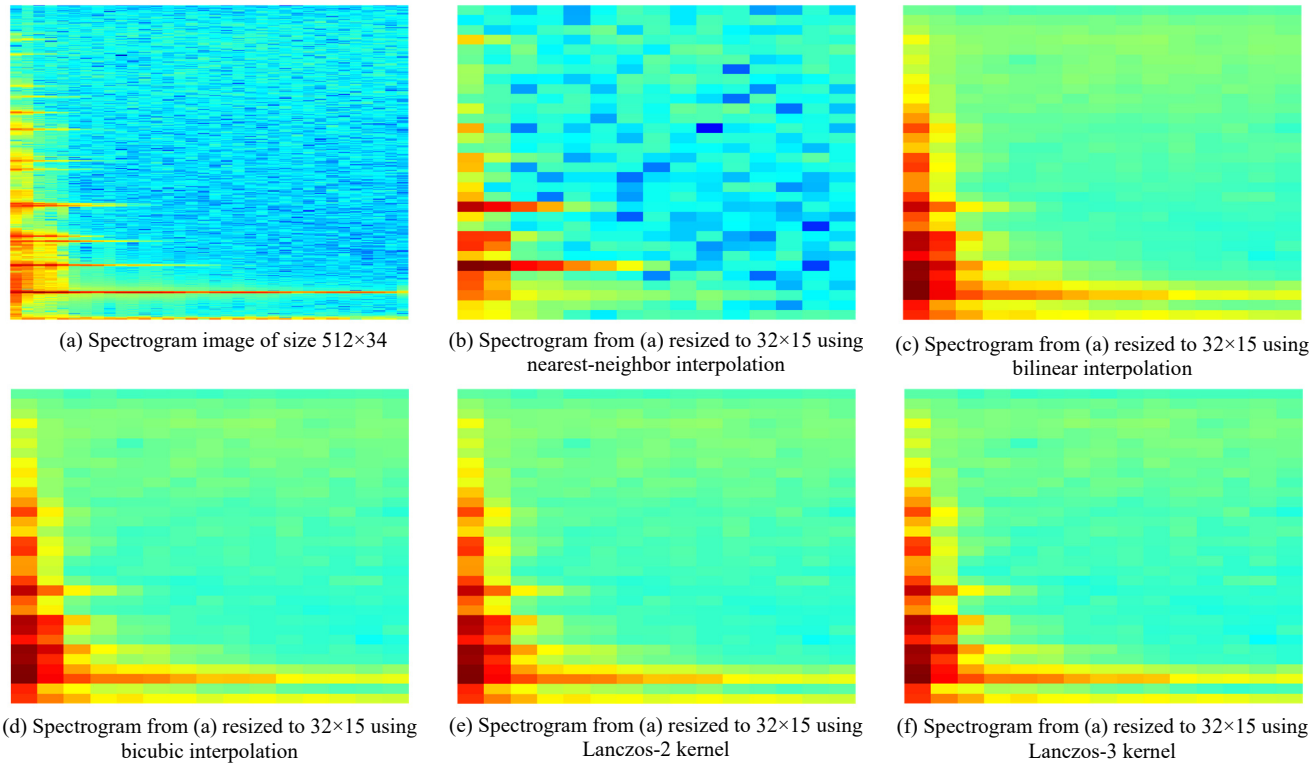
Results using the conventional spectrogram image and the various resized spectrograms with CNN classification are given in Table I. A baseline classification accuracy of 93.46% is achieved using the conventional spectrogram image. However, marginally to significantly better classification accuracy could be achieved when using the resized spectrogram as input image to the CNN classifier. The increase in classification accuracy over the baseline method are 0.41%, 3.29%, 3.67%, 3.62%, and 3.65% using nearest-neighbor, bilinear, bicubic, Lanczos-2, and Lanczos-3 interpolation methods, respectively.

The average RMSE for the image interpolation methods considered in this work are given in Table II. The procedure described in [12] was used in computing the RMSE. The smallest average RMSE is achieved using Lanczos-3 followed by Lanczos-2 and bicubic interpolation. Bicubic and Lanczos kernel interpolation methods have also been seen to produce low RMSE in digital image scaling applications, such as [13]. A classification accuracy of 97.13%, 97.08%, and 97.11% is achieved using bicubic, Lanczos-2, and Lanczos-3 kernel resized spectrograms, respectively, which are deemed to be the best suited for the application considered here.

In [3], using the same dataset and CNN architecture, a classification accuracy of 96.34%, 95.35%, and 98.03% was achieved with smoothed spectrogram, mel-spectrogram, and cochleagram, respectively. As such, the time-frequency representation formed using bicubic, Lanczos-2, and Lanczos-3 interpolation methods exceed the classification accuracy achieved using smoothed spectrogram and mel-spectrogram and are only marginally lower than the cochleagram representation. In addition, the resized spectrograms have an advantage in that scaling is done in both time and frequency dimensions eliminating the need for fixed number of frames and FFT points.

IV. Conclusion

Five image resizing techniques, nearest-neighbor interpolation, bilinear interpolation, bicubic interpolation, Lanczos-2 kernel interpolation, and Lanczos-3 kernel interpolation, are considered for time-frequency image resizing in acoustic event recognition using CNN. With a classification accuracy of over 97%, best results were achieved using spectrogram images resized using bicubic and Lanczos kernel interpolation methods. These results are



(a) Spectrogram image of size 512×34

(b) Spectrogram from (a) resized to 32×15 using nearest-neighbor interpolation

(c) Spectrogram from (a) resized to 32×15 using bilinear interpolation

(d) Spectrogram from (a) resized to 32×15 using bicubic interpolation

(e) Spectrogram from (a) resized to 32×15 using Lanczos-2 kernel

(f) Spectrogram from (a) resized to 32×15 using Lanczos-3 kernel

Fig. 3. (a) Spectrogram image of size 512×34 using frame length of 1024 points, 50% overlap between frames, and 1024 FFT points. The number of frames in this case is 34 but will differ for sound events with different signal length. Spectrogram from (a) resized to 32×15 using (b) nearest-neighbor interpolation, (c) bilinear interpolation, (d) bicubic interpolation, (e) Lanczos-2 kernel interpolation, and (f) Lanczos-3 kernel interpolation.

TABLE I. CLASSIFICATION RESULTS USING CONVENTIONAL AND VARIOUS RESIZED TIME-FREQUENCY IMAGE REPRESENTATIONS WITH CNN.

Time-frequency representation	**Classification Accuracy**
Spectrogram	93.46
Resized spectrogram (nearest-neighbor)	93.87
Resized spectrogram (bilinear)	96.75
Resized spectrogram (bicubic)	**97.13**
Resized spectrogram (Lanczos-2)	97.08
Resized spectrogram (Lanczos-3)	97.11

TABLE II. AVERAGE RMSE FOR THE DIFFERENT IMAGE INTERPOLATION METHODS APPLIED TO THE TIME-FREQUENCY IMAGES.

Image interpolation method	**RMSE**
Nearest-neighbor	4.7740
Bilinear	3.5500
Bicubic	3.4338
Lanczos-2	3.4316
Lanczos-3	**3.3957**

also better than what was achieved using smoothed spectrogram and mel-spectrogram in earlier work and only marginally lower than the results using cochleagram representation. In addition, the proposed methods have an advantage that there is no need for fixed number of frames and FFT points in computing the time-frequency image.

REFERENCES

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097-1105.

[2] O. Abdel-Hamid, A. R. Mohamed, H. Jiang, L. Deng, G. Penn, and D. Yu, "Convolutional neural networks for speech recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing,* vol. 22, no. 10, pp. 1533-1545, 2014.

[3] R. V. Sharan and T. J. Moir, "Acoustic event recognition using cochleagram image and convolutional neural networks," *Applied Acoustics,* vol. 148, pp. 62-66, 2019.

[4] J. Salamon and J. P. Bello, "Deep convolutional neural networks and data augmentation for environmental sound classification," *IEEE Signal Processing Letters,* vol. 24, no. 3, pp. 279-283, 2017.

[5] H. Zhang, I. McLoughlin, and Y. Song, "Robust sound event recognition using convolutional neural networks," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 559-563.

[6] I. Ozer, Z. Ozer, and O. Findik, "Noise robust sound event classification with convolutional neural network," *Neurocomputing,* vol. 272, pp. 505-512, 2018.

[7] C. F. Stallmann and A. P. Engelbrecht, "Signal modelling for the digital reconstruction of gramophone noise," in *International Conference on E-Business and Telecommunications (ICETE) 2015* Colmar, France, 2016, pp. 411-432.

[8] R. Keys, "Cubic convolution interpolation for digital image processing," *IEEE Transactions on Acoustics, Speech, and Signal Processing,* vol. 29, no. 6, pp. 1153-1160, 1981.

[9] W. B. Gearhart and H. S. Shultz, "The function sin x/x," *The College Mathematics Journal,* vol. 21, no. 2, pp. 90-99, 1990.

[10] K. Turkowski, "Filters for common resampling tasks," in *Graphics Gems*, A. S. Glassner, Ed. San Diego: Morgan Kaufmann, 1990, pp. 147-165.

[11] S. Nakamura, K. Hiyane, F. Asano, T. Nishiura, and T. Yamada, "Acoustical sound database in real environments for sound scene understanding and hands-free speech recognition," in *Proceedings of the 2nd International Conference on Language Resources and Evaluation (LREC 2000)*, Athens, Greece, 2000, pp. 965–968.

[12] A. Amanatiadis and I. Andreadis, "A survey on evaluation methods for image interpolation," *Measurement Science and Technology,* vol. 20, no. 10, p. 104015, 2009.

[13] P. Getreuer, "Linear methods for image interpolation," *Image Processing On Line,* vol. 1, pp. 238-259, 2011.