BLIND REVERBERATION TIME ESTIMATION USING A CONVOLUTIONAL NEURAL NETWORK

Hannes Gamper and Ivan J. Tashev

Audio and Acoustics Research Group
Microsoft Research
Redmond, WA, USA

ABSTRACT

The reverberation time of an acoustic environment is one of the most important parameters describing an environment’s acoustic behaviour. It is typically defined as the time, $T_{60}$, it takes for the acoustic impulse response (AIR) energy to decay by 60 dB. Besides its perceptual relevance [1, 2, 3], the $T_{60}$ is important in practical applications, including mixed reality and voice-controlled systems, as it affects the performance of sound source localisation [4] and speech recognition systems [5, 6].

While well-established methods exist to determine the $T_{60}$ from an AIR [7, 8], the AIR itself is typically not available in practical scenarios. Instead, the $T_{60}$ has to be inferred directly from signals present in the acoustic environment, e.g., speech captured by a user’s device. This can be challenging, especially if the recorded signals stem from unknown sources and are corrupted by ambient noise. In 2015 the ACE Challenge workshop was held to address the question of blindly estimating the $T_{60}$ from speech signals recorded in reverberant, noisy environments [9]. The challenge resulted in a number of state-of-the-art $T_{60}$ estimation methods, including classic signal processing as well as machine learning approaches.

Prego et al. contributed the method with the best performance in terms of the Pearson correlation coefficient between estimated and ground-truth $T_{60}$ [10, 11]. The method relies on estimating the signal-to-noise ratio (SNR) from silence at the beginning of a sample and applying a noise reduction technique before estimating the $T_{60}$ from features in the Short Time Fourier Transform (STFT) domain.

Faraji et al. propose fitting a first-order infinite impulse response (IIR) model to reverberant speech, showing a relation between the IIR pole and the $T_{60}$ [12]. While they report that their approach outperforms one of the baseline methods in the ACE Challenge [13], it is not clear how it would compare to state-of-the-art approaches. Lee and Chang employ a fully-connected neural network with three hidden layers to estimate the $T_{60}$ of speech samples convolved with synthetic room impulse responses and additive babble noise at 20 dB SNR [14]. They propose using a Mel-frequency spectrogram rather than STFT features to reduce computational complexity. However, experimental results are only reported for synthetic data, making it difficult to assess how well the method would generalise to real recordings of noisy, reverberant speech. Senoussaoui et al. propose an approach that fuses short- and long-term speech features to estimate $T_{60}$ from speech in noisy environments [15]. They show that their method outperforms one of the ACE Challenge baseline methods [16]. However, no comparison to state-of-the-art methods is reported. Lee and Chang propose using a deep neural network to estimate reverberation time from multi-channel recordings [17]. However, the method is only evaluated on simulated AIRs. The number, variety, and complexity of the previously proposed algorithms are an indication of the difficulty of the $T_{60}$ estimation problem.

Here we propose a single-channel $T_{60}$ estimation approach that is conceptually straightforward and computationally efficient, and we evaluate it on the ACE Challenge corpus, allowing direct comparison with state-of-the-art methods. Recently, convolutional neural networks have been applied successfully in many areas, including image classification [18], classic speech and audio problems [19, 4], as well as perceptual modelling [20]. They can be advantageous over standard feedforward networks as they have fewer free parameters and are thus easier to train [18]. We hypothesise that learning and combining convolution kernels is a well-suited approach for extracting features related to the $T_{60}$ from reverberant speech, as it enables the network to learn representations that rely both on local spectral and temporal features (e.g., short, band-limited decay slopes) and their combinations at higher abstraction levels.

2. REGRESSION LEARNING USING A NEURAL NETWORK

Machine learning has rapidly advanced the state-of-the-art in areas including speech recognition [21] and image classification [22]. In the audio domain, neural networks have been used successfully to estimate the $T_{60}$ [23], the direction of arrival of a sound source [4], and the polar angle of a binaural signal [20], by formulating the goal of estimating a continuous variable as a classification problem. Here we propose formulating the $T_{60}$ estimation as a regression problem. This has three advantages over using a classification approach:

- the ground truth $T_{60}$ data do not need to be quantised;
- we utilise a loss function that directly minimises the estimation error, rather than the classification error, potentially leading to better estimation performance [24];
- the model directly outputs a continuous-valued $T_{60}$ estimate.

978-1-5386-8151-0/18/$31.00 ©2018 IEEE
We use a convolutional neural network combined with a fully connected layer with a single output node that directly estimates $T_{60}$. As the training loss function we use the squared error between the $T_{60}$ estimate and the ground truth value. Unlike loss functions used for classification tasks, e.g., the cross-entropy loss, that do not encode class order or distance, the squared error is a distance metric. To minimise the training loss the network is forced to learn a representation that minimises the distance between samples with similar $T_{60}$. We hypothesise that this leads to a more robust model and better estimation performance than approaches based on classification.

### 3. PROPOSED APPROACH

The proposed approach for estimating $T_{60}$ blindly from speech relies on a deep convolutional network trained with a large number of noisy, reverberant speech samples with known $T_{60}$ values.

#### 3.1. Training data generation

Obtaining training data of sufficient quality and quantity is crucial for the performance of a deep neural network. To generate noisy, reverberant speech samples we follow the specifications for noise types and SNR levels outlined in the ACE Challenge [9]. To ensure that the trained model does not overfit the training data or hone in on artifacts stemming from the data synthesis process, it is important to carefully separate training and test data. A typical setup splits all available data into three separate sets: a training set for training the network; a validation set to monitor whether the network is overfitting during training; and a test set to evaluate the final trained model.

The data corpus used for the ACE Challenge contains separate development and evaluation sets [9]. Here we use the ACE development set for model validation, and the ACE evaluation set for model testing. To create training data, we generated 670 synthetic AIRs using the image source method for shoebox environments of varying sizes and with varying absorption coefficients [25]. For each simulated shoebox, one randomly placed source and five randomly placed receivers were simulated. The synthetic AIRs were combined with 655 measured multi-channel AIRs from internal and publicly available databases, including the Openair database [26], the RWCP database [27], the REVERB Challenge dataset [5], the EchoThief database [28], and datasets available in the SOFA format [29], yielding a total of 1325 AIRs with $T_{60}$ values between 0.1 and 1.5 s. The ground truth $T_{60}$ values were estimated using a method proposed by Karjalainen et al. [8]. A histogram of the $T_{60}$ values of all AIRs used to generate training data is shown in Figure 1. As can be seen, the distribution is not uniform, and it is not clear whether this distribution is representative of the $T_{60}$ values encountered in real environments. For methods to address class imbalance with convolutional neural networks the reader is referred to the work by Buda et al. [30]. No such methods were considered in the present work.

Speech samples were selected randomly from the TIMIT database [31] and an internal corpus of close-mic recordings. After discarding samples with low SNR or other artifacts, this resulted in a set of 903 English utterances by female and male speakers.

For the purposes of this work we only considered the noise types contained in the ACE corpus: “ambient”, “fan”, and “babble” [9]. Due to this limitation, it is not clear how the proposed approach would behave in the case of noise types not seen during training. To simulate ambient and fan noise, we extracted the magnitude spectra of random 10 s long segments of the corresponding noise recordings in the ACE development set and shaped white Gaussian noise with those spectra. 22 anechoic sound samples (foot steps, coughing, office equipment, etc.) were randomly added to the noise samples to simulate non-stationary background noise. We then convolved these noise samples with the multi-channel AIRs in our training set to simulate decorrelated ambient and fan noise recordings. To simulate babble noise, we convolved random speech samples from our speech corpus with the multi-channel AIRs and added them to the simulated ambient noise samples.

The training samples were created by convolving random speech samples with the training AIRs and adding the synthetic noise samples to yield SNRs of 0, 10, and 20 dB, using the tools provided with the ACE corpus [9]. This resulted in a total of 23,850 reverberant, noisy training utterances with durations between 4 and 10 s. Table 1 summarises the data sets used in this work.

#### 3.2. Data preprocessing

While neural network architectures exist that consume raw audio [19], and the first layers of deep convolutional neural networks have been shown to learn pre-processing filters directly from the
The convolutional neural network was implemented using the Microsoft Cognitive Toolkit (CNTK) [34] and trained using the data generated as described in Section 3.1. Training was performed on two GPUs using stochastic optimisation [35] of a squared error loss function, over 1500 epochs. The final trained model was tested using the evaluation set of the ACE Challenge corpus [9].

To evaluate the performance of the proposed method the same criteria as for the ACE Challenge are used [9]:

- the estimation bias, calculated as the mean of the estimation error;
- the mean squared error (MSE);
- the Pearson correlation coefficient, $\rho$, between estimated and ground truth $T_{60}$ values.

The ACE Challenge results also include the real-time factor (RTF) for each algorithm, calculated as the ratio between compute time and sample duration [11]. However, as the compute time is dependent on the specific computer hardware used for evaluation, the RTF of the proposed method and those reported in the ACE Challenge are not directly comparable. For reference, the RTF of the proposed method was estimated at about 0.05, i.e., 20 times faster than real-time, with about 95% of the compute time spent pre-processing speech files in Matlab. The actual $T_{60}$ estimation using the pre-processed input samples and running on a GPU clocked in at a RTF of about 0.002, i.e., 500 times faster than real-time, due to CNTK being highly optimised [34]. While this result is not directly comparable to RTF values reported for the algorithms in the ACE Challenge, it serves as an indication of the computational efficiency of the proposed method.

Table 2 shows a comparison of the proposed method and two benchmark algorithms from the ACE Challenge: the best-performing machine-learning based algorithm [23], as well as the best-performing algorithm overall [10]. As can be seen, the proposed method outperforms both benchmark algorithms on all metrics, achieving the lowest bias and MSE as well as the highest Pearson correlation coefficient, $\rho$. It should be noted that the proposed method operates on input chunks with a fixed length of four seconds, i.e., shorter utterances are zero-padded while longer

<table>
<thead>
<tr>
<th></th>
<th>conv1</th>
<th>conv2</th>
<th>conv3</th>
<th>conv4</th>
<th>conv5</th>
<th>conv6</th>
</tr>
</thead>
<tbody>
<tr>
<td>size</td>
<td>1 x 10</td>
<td>1 x 10</td>
<td>1 x 11</td>
<td>1 x 11</td>
<td>3 x 8</td>
<td>4 x 7</td>
</tr>
<tr>
<td>stride</td>
<td>(1, 2)</td>
<td>(1, 3)</td>
<td>(1, 3)</td>
<td>(1, 2)</td>
<td>(2, 2)</td>
<td>(2, 1)</td>
</tr>
</tbody>
</table>
utterances result in multiple $T_{60}$ estimates. Although measures were taken to prevent overfitting, the performance of the model on the training set was substantially better than on the ACE Challenge evaluation set, with a bias of 0.0055, a MSE of 0.0125, and a Pearson correlation coefficient of 0.953. This discrepancy between training and test performance illustrates the importance of strictly separating training and test sets when evaluating a machine learning model, especially when using small and/or synthetic data sets, as is quite common in the audio domain. Figure 4 shows the error performance for the training set (top) and the ACE Challenge evaluation set (bottom). The estimation error seems to increase towards higher $T_{60}$ values. This is expected, as estimating a long $T_{60}$ presumably requires a long input sample that is sufficiently sparse to observe long energy decays. Figure 5 shows confusion matrices for the training set (left) and the ACE Challenge evaluation set (right). As can be seen, the estimation error is distributed around the true $T_{60}$ value, indicating that the CNN successfully learned a representation of the underlying regression problem. SNR did not seem to have a major effect on performance for the noise levels and types tested here.

5. CONCLUSION AND FUTURE WORK

Blind $T_{60}$ estimation from noisy, reverberant speech remains a challenging problem. We show that the estimation can be modelled as a regression problem and implemented with a convolutional neural network (CNN). After training the model using over 30,000 input samples containing varying levels of ambient noise and reverberation and taking measures to combat overfitting, the proposed method is shown to outperform state-of-the-art algorithms on the ACE Challenge evaluation corpus for $T_{60}$ estimation [9, 11]. Due to the highly optimised implementation of the model [34], the proposed estimation algorithm is shown to be computationally efficient, running significantly faster than real-time.

While these results are encouraging and prove the potential of the proposed approach, future work is needed to expand the training set, e.g., by collecting more data or using data augmentation [18]. This would potentially allow training a more complex network architecture with higher capacity and better performance. Furthermore, the model should be evaluated on a broader set of test cases to verify that the method generalises to unseen scenarios and noise types.

6. REFERENCES


A. Politis, Microphone array processing for parametric spatial audio techniques, G5 artikkelivitskirja, Aalto University, 2016.


