Supplementary Materials: The Boombox: Visual Reconstruction from Acoustic Vibrations

Anonymous Author(s)
Affiliation
Address
`email`

1 Data Processing Details

There are three modalities (RGB, depth, and audio) in our collected data. Our goal for data processing is to obtain high-quality data samples for coherent dynamics and scene representation.

For each video recording, we aligned the RGB and depth streams with the camera parameters and extracted the last frame to represent the final scene after the object became stable. We applied edge-preserving and hole-filling spatial filters on the depth images to get less noisy depth images. We then cropped and resized all the image frames to remove the unnecessary backgrounds. In order to quantitatively evaluate our predictions, we further applied background subtractions leading to binary object masks.

Audio clips were recorded continuously across per data collection trail which included multiple sequences. Therefore, the first step requires segmenting the audio clips for each dropping sequence. We first calculate the audio segments for each microphone with energy threshold. As a means to synchronize among all four microphones, we use the earliest and latest sound arrival timestamps from four audio segments on the same sequence to finalize the segmentation windows.

2 Network Architectures

We list all the specific parameters of the encoder and decoder network in Table 1 and Table 2. All layers are accompanied with a batch normalization layer and a specified activation function. We will also open source all the hardware and software design along with the collected dataset.

Submitted to the 5th Conference on Robot Learning (CoRL 2021). Do not distribute.

Layer	Kernel Size	#Filters	Stride	Padding	Dilation	Activation
Conv1	4x4	32	2	1	1	ReLU
Conv2	4x4	32	2	1	1	ReLU
Conv3	4x4	64	2	1	1	ReLU
Conv4	4x4	128	2	1	1	ReLU
Conv5	4x4	128	2	1	1	ReLU
Conv6	4x4	128	2	1	1	ReLU
Conv7	4x4	128	2	1	1	ReLU

Table 1: **Encoder Network Parameters:** we list all the specific parameters for the encoder network. In addition to the above convolutional layers, after each "Conv" layer, we attach another convolutional layer with the same number of filters as the current convolutional layer but with 4×4 kernel and 3 as stride.

Layer	Kernel Size	#Filters	Stride	Padding	Dilation	Activation
Deconv7	4x4	128	2	1	1	ReLU
Deconv6	4x4	128	2	1	1	ReLU
Deconv5	4x4	128	2	1	1	ReLU
Deconv4	4x4	64	2	1	1	ReLU
Deconv3	4x4	32	2	1	1	ReLU
Deconv2	4x4	16	2	1	1	ReLU
Deconv1	4x4	3/1	2	1	1	Sigmoid

Table 2: **Decoder Network Parameters:** we list all the specific parameters for the decoder network. Except for the last layer, along with every transposed convolutional layer, the input of each layer is also first passed through a 2D convolutional layer with kernel size 3×3 and stride 1 and then a transposed 2D convolutional layer with kernel size 3×3 and stride 2. The output of this branch will then be concatenated with each "Deconv" layer along the feature dimension as the input of the next "Deconv" layer.