APPENDIX

A. Sabine Formula

1 second audio frames was based on the Sabine Formula of reverberation time for a compact room of like dimensions. For the bathroom scene, $V = 9\ ft * 16\ ft * 9\ ft = 1,296\ ft^3$ and $a = 69.23\ ft^2$, which is the sum of sound absorption from the materials in Table (I).

TABLE I

SABINE FORMULA : REVERBERATION TIME CAN BE CALCULATED AS ROOM VOLUME V DIVIDED BY TOTAL ROOM ABSORPTION A. FOR AN INDOOR SOUND SOURCE IN A REVERBERANT FIELD, A IS THE TOTAL ROOM ABSORPTION AT A GIVEN FREQUENCY (SABINS), S IS THE SURFACE AREA (ft^2), AND α IS THE SOUND ABSORPTION COEFFICIENT AT A GIVEN FREQUENCY (DECIMAL PERCENT).

Total room absorption a using $a = \sum S\alpha$ at 250 Hz			
Real bathroom scene	S	α	a (sabins)
Painted walls	432 x	0.10 =	43.20
Tile floor	175 x	0.01 =	1.75
Glass	60 x	0.25 =	15.00
Ceramic	39 x	0.02 =	0.78
Mirror	34 x	0.25 =	8.50

B. Synthetic Data

The automated synthetic data collection was performed in Unreal Engine 4.25, where SteamAudio employs a ray-based geometric sound propagation approach, with support for dynamic geometry using the Intel Embree CPU-based ray-tracer. We describe similar prior work GSoundhere for more details on this approach. Given scene materials (e.g. carpet, glass, painted, tile, etc.), a sound source (e.g. voice), environmental geometry, and listener position, we generate impulse responses for a given scene of varying sizes. From each listener, specular and diffuse rays are randomly generated and traced into the scene. The energy-time curve for simulated impulse response $S_f(t)$ is the sum of these rays:

$$S_f(t) = \sum_j \delta(t - t_j) I_{j,f} \tag{1}$$

where $I_{j,f}$ is the sound intensity for path j and frequency band f, t_j is the propagation delay time for path j, and $\delta(t - t_j)$ is the Dirac delta function or impulse function. As sound rays collide in the scene, their paths change based on absorption and scattering coefficients of the colliding objects [**?**]. We assume a sound absorption coefficient, $\alpha = 1.0$ for open windows.

Along with sound intensity $S_f(t)$, a weight matrix W_f is computed on materials within the scene. Each entry $w_{f,m}$ is the average number of reflections from material m for all paths that arrived at the listener. It is defined as:

$$w_{f,m} = \frac{\sum I_{j,f} d_{j,m}}{\sum I_{j,f}} \tag{2}$$

where $d_{j,m}$ is the number of times rays on path j collide with material m, weighted according to sound intensity $I_{j,f}$ of the path j. To mirror real-world data, sound source directivity was disabled.

Given a 720p 30fps video walkthrough of the 3D environment with the camera moving along a keyframed spline, we reconstruct the virtual scene by extracting the individual frames of the video and using Agisoft Metashape (v1.7)'s reconstruction pipeline to solve for each image's camera transform. Metashape, previously known as PhotoScan, is considered state-of-the-art in commercial photogrammetry software. The general process is: create a sparse point cloud containing only keypoints and solve for the transforms of cameras that can see the keypoints, create a dense cloud by matching more features between keypoint-seeing cameras and the rest, project the dense cloud depth data to each camera to build per-camera depth maps, use the depth maps to build a mesh, and create a texture map by projecting the image frames that best see each polygon onto the mesh.

We disable motion blur of the camera in order to have more usable frames, but real camera data generally requires blurry frames to be removed to avoid noisy reconstructions, especially at low framerates. For accurate visual feedback of the specular surfaces, we also enable UE4's DirectX12 ray-tracing for reflective and translucent surfaces. We used a PC with the following specs for reconstruction: GTX 1080 GPU, i9-9900k CPU, 64gb RAM, Windows 10 x64, taking about 2 hours to process a 720p sequence of 2,000 images from start to finish with this setup. We use three sections of the "HQ Residential House" environment on the Unreal Marketplace for synthetic data. The kitchen and bathroom result in about 2,000 images when extracted from the video at a step size of 3, and the master bedroom about 4,000.

C. Scene and Audio Reconstruction for VR Systems

When using a head mounted display (HMD) users are alerted when approaching the boundaries in physical space. However, if room setup does not accurately reflect these boundaries or changes occur after setup, a user risks walking into unseen real-world objects such as glass and walls. Using our method, transmitted sound from the HMD could be used to locate physical objects and appropriately notify the user as an added safety measure. Audio directly from the real-world environment could also be used for depth estimation. The sounds unmixed and placed in the virtual environment, reconstructing both the scene geometry and sound sources (Fig. 2). Finally, seasonal variations in the 3D sound and visual reconstruction of a window open in the spring and closed in the winter also enhance the AR/VR experience. See supplementary demo video.

We evaluated the smartphone based reconstruction applications to obtain an initial 3D geometry for which our method would enhance. Astrivis application generates better 3D geometries for closed object rather than scene reconstructions, since it limits feature points per scan. On the other hand, Agisoft Metashape produces scene reconstructions offline from smartphone video. Enabling the software's depth point and guided camera matching features further improved reconstructed geometries.

Fig. 1. Listener at different distances from sound source (from 0.5 to 3 m) in a virtual environment (left: bathroom, middle: kitchen, right: bedroom) used to generate synthetic audio-visual data. This dataset is comprised of multiple 12-second video clips in front of reflective surfaces at increments from 0.5 m to 3 m for 15 different sound sources. Absorption and transmission coefficients were set on materials (e.g. mirror, thick glass, ordinary glass) inside and outside of rooms in the virtual scenes. These scenes are used in controlled experiments summarized in Table II.

Accuracy of Reflecting Sounds used for Classification in Controlled Experiment (Gl = Glass)

Method	Input	Open/Closed		Depth Est. (+/- 0.5 m)		Sound
		Thick Gl	Thin Gl	Thick Gl	Thin Gl	Material Est
kNN [**?**]	A	53.8%	64.1%	11.5%	21.4%	66.5%
Linear SVM [**?**]	A	54.7%	63.2%	11.5%	20.5%	61.1%
SoundNet5 [**?**]	A	60.0%	40.1%	18.8%	19.1%	67.4%
SoundNet8 [**?**]	A	60.0%	42.6%	25.0%	19.1%	34.0%
EchoCNN-A (Ours)	A	**61.1%**	**65.1%**	**44.4%**	**44.6%**	**68.1%**
AlexNet [**?**]	V	95.8%	80.8%	83.3%	66.7%	87.5%
Acoustic Classification [**?**]	AV	N/A	N/A	N/A	N/A	– 48% * –
EchoCNN-AV Cat (Ours)	AV	98.9%	**100%**	99.4%	92.2%	76.6%
EchoCNN-AV MFB (Ours)	AV	**100%**	**100%**	**100%**	**99.0%**	**100%**

TABLE II

MULTIPLE MODELS (**OURS IS ECHOCNN**) AND BASELINES WERE EVALUATED FOR AUDIO AND AUDIO-VISUAL BASED SCENE RECONSTRUCTION ANALYSIS IN CONTROLLED EXPERIMENTS OF VIRTUAL ENVIRONMENTS.

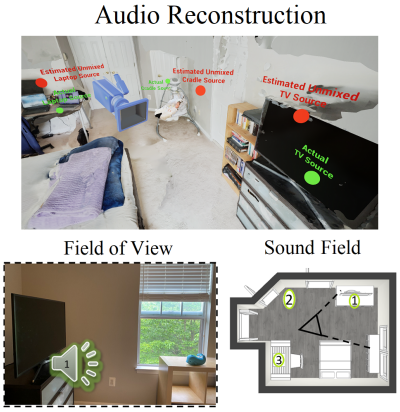

Fig. 2. EchoCNN may also be used to reconstruct the audio of a virtual scene from a video of a room in a real scene. Instead of depth estimation, our method can be trained to approximate sound source position, which is especially useful for objects that are outside of the camera field of view. Ground truth (green dots) and estimated (red dots) sound source placements are shown (top). Seen and heard sound source (TV) from the video capture is placed more accurately than unseen but heard sound sources (cradle and laptop).

D. Results by Source Frequency and Object Size

We evaluate a range of source frequencies to account for different sound wave behavior based on the size of the reconstructing objects. For example, if an object is much smaller than the wavelength, the sound flows around it rather than scattering [**?**]. Dynamically setting source frequency based on object size could use $\lambda = \frac{c}{f}$ where λ is wavelength (ft) of sound in air at a specific frequency, f is frequency (1 Hz), and c is speed of sound (ft/s).

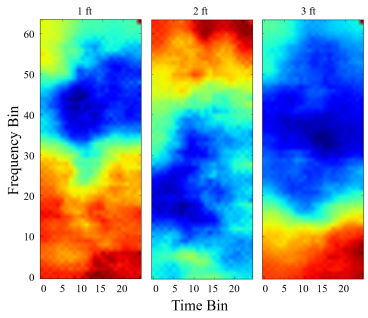

Fig. 3. *Left to right*: audio input produces the highest activation for a given depth class from 1 ft, 2 ft, and 3 ft away from an object. Longer reverberation times tend to occur at lower frequencies (3 ft) than at high frequencies (1 and 2 ft) due to typical high frequency damping and absorption.

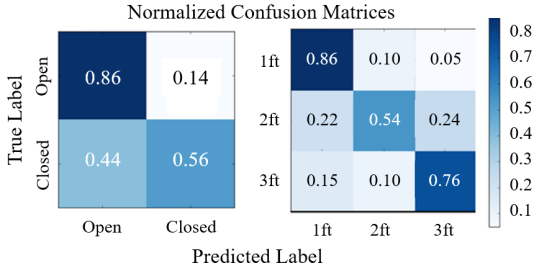

Fig. 4. EchoCNN-A (*Left*) Confusion matrix to classify open/closed for an interior glass shower door. Open predictions (86%) were more accurate than closed (56%). (*Right*) Confusion matrix to classify depth from same interior glass door. Notice that our EchoCNN is learning to differentiate distance based on reflecting sounds from pulsed ambient waves of a smartphone.