

# MULTIMODAL LEARNING FOR AUDIO AND VISUAL PROCESSING

Justin Alden Wilson

A dissertation submitted to the faculty at the University of North Carolina at Chapel Hill in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Computer Science.

Chapel Hill  
2020

Approved by:

Henry Fuchs

Gary Bishop

Ming Lin

Dinesh Manocha

Shahriar Nirjon

© 2020  
Justin Alden Wilson  
ALL RIGHTS RESERVED



## ABSTRACT

Justin Alden Wilson: Multimodal Learning for Audio and Visual Processing  
(Under the direction of Henry Fuchs and Ming C. Lin)

The world contains vast amounts of information which can be sensed and captured in a variety of ways and formats. Virtual environments also lend themselves to endless possibilities and diversity of data. Often our experiences draw from these separate but complementary parts which can be combined in a way to provide a comprehensive representation of the events. Multimodal learning focuses on these types of combinations. By fusing multiple modalities, multimodal learning can improve results beyond individual mode performance. However, many of today’s state-of-the-art techniques in computer vision, robotics, and machine learning rely solely or primarily on visual inputs even when the visual data is obtained from video where corresponding audio may also be readily available to augment learning. Vision only approaches can experience challenges in cases of highly reflective, transparent, or occluded objects and scenes where, if used alone or in conjunction with, audio may improve task performance. To address these challenges, this thesis explores coupling multimodal information to enhance task performance through learning-based methods for audio and visual processing using real and synthetic data.

Physically-based graphics pipelines can naturally be extended for audio and visual synthetic data generation. To enhance the rigid body sound synthesis pipeline for objects containing a liquid, I used an added mass operator for fluid-structure coupling as a pre-processing step. My method is fast and practical for use in interactive 3D systems where live sound synthesis is desired. By fusing audio and visual data from real and synthetic videos, we also demonstrate enhanced processing and performance for object classification, tracking, and reconstruction tasks.

As has been shown in visual question and answering and other related work, multiple modalities have the ability to complement one another and outperform single modality systems. To the best of my knowledge, I introduced the first use of audio-visual neural networks to analyze liquid pouring sequences by classifying their weight, liquid, and receiving container. Prior work often required predefined source weights or visual data. My contribution was to use the sound from a pouring sequence—a liquid being poured

into a target container- to train a multimodal convolutional neural networks (CNNs) that fuses mel-scaled spectrograms as audio inputs with corresponding visual data based on video images.

I described the first use of an audio-visual neural network for tracking tabletop sized objects and enhancing visual object trackers. Like object detection of reflective surfaces, object trackers can also run into challenges when objects collide, occlude, appear similar, or come close to one another. By using the impact sounds of the objects during collision, my audio-visual object tracking (AVOT) neural network can correct trackers that drift from their original objects that were assigned before collision.

Reflective and textureless surfaces not only are difficult to detect and classify, they are also often poorly reconstructed and filled with depth discontinuities and holes. I proposed the first use of an audio-visual method that uses the reflections of sound to aid in geometry and audio reconstruction, referred to as *"Echoreconstruction"*. The mobile phone prototype emits pulsed audio, while recording video for RGB-based 3D reconstruction and audio-visual classification. Reflected sound and images from the video are input into our audio (EchoCNN-A) and audio-visual (EchoCNN-AV) convolutional neural networks for surface and sound source detection, depth estimation, and material classification. EchoCNN inferences from these classifications enhance scene 3D reconstructions containing open spaces and reflective surfaces by depth filtering, inpainting, and placement of unmixed sound sources in the scene.

In addition to enhancing scene reconstructions, I proposed a multimodal single- and multi-frame reconstruction LSTM autoencoder for 3D reconstructions using audio-visual inputs. Our neural network produces high-quality 3D reconstructions using voxel representation. It is the first audio-visual reconstruction neural network for 3D geometry and material representation.

Contributions of this thesis include new neural network designs, new enhancements to real and synthetic audio-visual datasets, and prototypes that demonstrate audio and audio-augmented performance for sound synthesis, inference, and reconstruction.

*To my wife, son, mom, dad, and all who believed in me*

## ACKNOWLEDGEMENTS

First and foremost, I would like to thank my advisors Henry Fuchs and Ming Lin, who have guided me and supported me throughout my PhD career. Thank you for your dedication to your students and contributions to computer science and affiliated disciplines. Your advising, teaching, and research has changed the world for the better. Without you, this dissertation would not exist. I also want to thank my committee members: Gary Bishop, Dinesh Manocha, and Shahriar Nirjon, for being a part of the research communities in visual reconstruction, sound simulation, and mobile devices. Your brilliant ideas continue to push the boundaries of research and inspire the direction of my research everyday.

Thanks to the GAMMA research group. I always looked forward to weekly meetings and emails to hear about the amazing research everyone was conducting. Special thanks to lab members, co-authors, and friends Auston Sterling, Autl Rungta, and Nick Rewkowski. You always made yourselves available in and outside of the lab to chat about research, lend listening ears and helping hands, grab me to play foosball, and share a smile.

Thanks to the Graphics & Virtual Reality research group and lab members. Special thanks to Alex Blate, Jim Mahaney, Praneeth Chakravarthula, Kishore Rathinavel, and Rohan Chabra for staying late to help with a demo or deadline and sharing their technical expertise in computer science and engineering.

I want to thank my wife, Seunghye Jung Wilson, and son, Gerard Shiwoo Wilson, for being with me throughout my journey. I love you infinite and always. Thanks to my parents who influenced me to work hard while staying compassionate.

I am grateful for the department's faculty, staff, and students. The graduate program is possible due to their efforts. I will also remember the times in graduate school bonding with friends: Tanya Amert, Tamzeed Islam, Bashima Islam, Ramakanth Pasunuru, Marc Eder, Adam Humphries, and many others.

Finally, thanks to the U.S. National Science Foundation and the Elizabeth Stevinson Iribe Chair Professorship for supporting parts of my research.

## TABLE OF CONTENTS

LIST OF TABLES .....	xii
LIST OF FIGURES .....	xiii
LIST OF ABBREVIATIONS .....	xvi
CHAPTER 1: INTRODUCTION .....	1
1.1 Motivation .....	1
1.2 Scope of this dissertation .....	7
1.3 Thesis Statement .....	7
1.4 Main Results .....	7
1.4.1 Sound Synthesis for Fluid-Structure Coupling.....	7
1.4.2 Analyzing Liquid Pouring Sequences via Audio-Visual Neural Networks.....	8
1.4.3 Audio-Visual Object Tracking of Multiple Objects .....	9
1.4.4 Audio-Augmented Scene and Object Reconstruction .....	9
1.5 Contributions of this dissertation .....	11
1.6 Organization .....	11
CHAPTER 2: SOUND SYNTHESIS FOR FLUID-STRUCTURE COUPLING .....	12
2.1 Introduction.....	12
2.1.1 Contributions.....	13
2.2 Related Work .....	14
2.3 System Overview .....	16
2.3.1 Rigid-Body Sound Synthesis Pipeline .....	16
2.3.2 Modal Sound Analysis.....	17
2.3.3 Modal Sound Synthesis .....	18

2.3.4	Added Mass Operator .....	18
2.4	Sound Synthesis for Fluid-Structure Coupling .....	20
2.4.1	Identifying Nodes at Fluid-Structure Interface .....	20
2.4.2	Modifying the Mass Matrix .....	21
2.4.3	Hydrostatic Force on a Curved Surface .....	22
2.5	Results and Analysis .....	24
2.5.1	Comparison: Synthesized vs. Recording .....	24
2.5.2	Spectrogram Analysis .....	26
2.6	Applications .....	27
2.6.1	Simulated Liquid Xylophone .....	28
2.6.2	Integration With Virtual Environments .....	28
2.7	Conclusion and Future Work .....	28
CHAPTER 3: ANALYZING LIQUID POURING SEQUENCES VIA AUDIO-VISUAL .....		31
3.1	Introduction .....	31
3.2	Related Work .....	33
3.3	Technical Approach .....	35
3.3.1	Task Overview .....	35
3.3.2	Audio Feature Analysis: Helmholtz Resonance Frequency .....	36
3.3.3	Dataset Generation .....	37
3.3.4	Neural Network Architecture of Audio-based Method .....	38
3.3.5	Neural Network Architecture of Audio-Visual Method .....	39
3.3.6	Implementation Details .....	40
3.4	Results .....	40
3.4.1	Data Capture and Training .....	41
3.4.2	Pouring Sequence Experiments .....	41
3.4.3	Our PSNN Accuracy vs. Baseline Results .....	42
3.4.4	Liquid and container classification .....	45

3.5	Analysis .....	46
3.5.1	Activation Maximization Visualizations .....	46
3.5.2	Model Comparisons .....	46
3.5.2.1	PSNN-A Normalized .....	47
3.5.2.2	PSNN-A and Temporal Convolutional Networks (TCN) .....	47
3.5.2.3	Robot and Human Poured .....	47
3.5.2.4	Combined Pour Dataset .....	47
3.5.2.5	Interval Length .....	48
3.6	Conclusion and Future Work .....	48
CHAPTER 4: AUDIO-VISUAL OBJECT TRACKING FOR MULTIPLE OBJECTS .....		50
4.1	Introduction .....	50
4.2	Background and Related Work .....	51
4.2.1	Object Detection .....	53
4.2.2	Object Tracking .....	53
4.2.3	Audio-Visual Methods .....	55
4.3	Technical Approach .....	56
4.3.1	AVOT Neural Network Architecture .....	57
4.3.2	AVOT Dataset .....	58
4.3.3	Implementation Details .....	59
4.4	Experiments and Results .....	60
4.4.1	Our Results vs. Baselines .....	61
4.4.2	Maximization Activation .....	62
4.5	Conclusion .....	62
CHAPTER 5: AUDIO-AUGMENTED SCENE RECONSTRUCTION ON MOBILE DEVICES ....		63
5.1	Introduction .....	63
5.2	Related Work .....	65
5.2.1	3D reconstruction .....	65

5.2.1.1	Glass and mirror reconstruction .....	66
5.2.2	Acoustic imaging and audio-based classifiers .....	66
5.3	Technical Approach .....	68
5.3.1	Echolocation .....	68
5.3.2	Staged classification and reconstruction pipeline .....	69
5.3.3	Sound source .....	69
5.3.4	Model Architecture .....	71
5.3.5	Loss Function .....	73
5.3.6	Depth filtering and planar inpainting .....	74
5.4	Datasets .....	74
5.4.1	Real and synthetic datasets .....	75
5.5	Experiments and Results .....	76
5.5.1	Experimental setup .....	77
5.5.2	Implementation details .....	77
5.5.2.1	Initial 3D Reconstruction .....	78
5.5.3	Results by source frequency and object size .....	78
5.5.4	Activation Maximization .....	79
5.5.5	Applications .....	79
5.6	Conclusion and Future Work .....	79
CHAPTER 6:	AUDIO-VISUAL OBJECT RECONSTRUCTION FROM VIDEO .....	83
6.1	Introduction .....	83
6.2	Related Work .....	85
6.2.1	3D Reconstruction .....	85
6.2.2	Multimodal Neural Networks .....	85
6.2.3	Reconstruction Network Structures .....	86
6.3	Technical Approach .....	87
6.3.1	Object Tracking and Visual Representation .....	87



6.3.2	Sound Source Separation of Impact Sounds and Audio Representation .....	88
6.3.2.1	Single View, Single Impact Sound .....	89
6.3.2.2	Multi-Frame, Multi-Impact.....	89
6.4	3D-MOV Network Structure .....	90
6.4.1	Single Frame Feature Extraction .....	91
6.4.2	Frame Aggregation .....	91
6.4.3	Modality Fusion and 3D Decoder .....	91
6.5	Results .....	92
6.5.1	Implementation.....	93
6.5.2	Training.....	93
6.5.3	Evaluation metrics .....	94
6.6	ML Reproducibility .....	94
6.6.1	Experimental Results .....	94
6.6.2	Datasets.....	95
6.7	Conclusions.....	95
CHAPTER 7: SUMMARY AND CONCLUSIONS.....		99
7.1	Summary of Results .....	99
7.2	Limitations and Future Work.....	101
7.3	Conclusion .....	103
REFERENCES .....		104

## LIST OF TABLES

Table 2.1 – Generalizations for different liquid densities comparing synthesized vs. actual frequencies	25
Table 2.2 – Generalizations for different liquid volumes comparing synthesized vs. actual frequencies	26
Table 2.3 – Solid and liquid tetrahedral meshes to modify mass elements of boundary nodes . . . . .	27
Table 2.4 – Generalizations for different solids and liquid volumes . . . . .	30
Table 3.1 – Ground truth and predicted labels for a pouring sequence with different time intervals . . .	37
Table 3.2 – Multiple models and baselines were evaluated for audio and audio-visual based liquid pouring analysis . . . . .	40
Table 3.3 – Evaluation results for the combined container dataset . . . . .	41
Table 3.4 – Various pouring contents were evaluated for weight estimation . . . . .	43
Table 3.5 – Multiple network models and baselines were evaluated for weight estimation of robot pouring sequences . . . . .	44
Table 3.6 – Varying type of liquid poured, multiple network models and baselines were eval- uated for weight estimation of robot pouring sequences. . . . .	44
Table 3.7 – PSNN-A predicts pouring content and target container . . . . .	45
Table 4.1 – Object detection and tracking datasets . . . . .	52
Table 4.2 – Multiple network models were evaluated on object tracking accuracy and time . . . . .	61
Table 5.1 – 3D reconstruction methods by type (active, passive, or other sensor), single or multi- view, and static or dynamic . . . . .	66
Table 5.2 – Reverberation time calculated for a room using the Sabine Formula . . . . .	71
Table 6.1 – Evaluated against baselines for loss and reconstruction accuracy . . . . .	95
Table 7.1 – Summary of contributions . . . . .	100

## LIST OF FIGURES

Figure 2.1 – Simulated liquid xylophone with varying levels of liquid and/or density .....	13
Figure 2.2 – A wooden pot, metallic teapot, and porcelain bowl with fine and coarse subdivision surfaces simulated with varying volumes of liquid .....	14
Figure 2.3 – Enhanced rigid body sound pipeline with new contributed steps to account for different liquid volumes and densities contained within the solid object .....	17
Figure 2.4 – Surface boundary and adjacent domain nodes are modified by an added mass based on the amount and type of liquid .....	21
Figure 2.5 – Added mass distributed along the fluid-structure interface .....	22
Figure 2.6 – Free-body diagram for the hydrostatic force along the fluid-structure boundary of a curved surface .....	23
Figure 2.7 – Increasing liquid volume in a wineglass decreases fundamental frequency resulting in a lower pitch .....	24
Figure 2.8 – Given equal volumes of liquid, the fundamental frequency of the sounding object decreases as the contained liquid density increases .....	25
Figure 2.9 – Fundamental frequency of the object decreases as the amount of liquid increases .....	26
Figure 2.10 – Empty wineglass time signal, power spectral density, and spectrogram .....	27
Figure 2.11 – Half full wineglass time signal, power spectral density, and spectrogram .....	28
Figure 2.12 – User strikes each glass interactively in a virtual environment using a mouse click or virtual reality controller to play an octave of a simulated liquid xylophone .....	29
Figure 2.13 – Users can interactively hear the sounds of various objects due to varying amounts and types of liquids in a virtual kitchen scene .....	29
Figure 3.1 – Our audio-augmented approach that performs weight estimation, overflow detection, and content and container classification .....	32
Figure 3.2 – Spectrogram from a recorded pouring sequence .....	34
Figure 3.3 – System overview for estimating poured amounts using audio-visual data from robot pouring sequences .....	36
Figure 3.4 – Audio-visual inputs of a mel-scaled spectrogram and cropped grayscale image .....	38

Figure 3.5 – Demo video of liquid weights predicted by our PSNN neural network for a robot pouring sequence .....	39
Figure 3.6 – Confusion matrix comparing actual to predicted poured amounts by classes of 0.2 oz (about 6 ml) weight increments .....	42
Figure 3.7 – Estimated weight accuracy within +/- 0.2 oz, 0.4 oz, and 0.6 oz of ground truth for pouring contents .....	43
Figure 3.8 – Confusion matrix of actual and predicted container classifications based on audio-only pouring sequences .....	45
Figure 3.9 – Audio and visual inputs that maximize Pouring Sequence Neural Network (PSNN) activations .....	46
Figure 4.1 – An example failure case of tracking colliding objects improved by our audio-visual object tracker .....	51
Figure 4.2 – Audio-Visual Object Tracker (AVOT) neural network architecture .....	54
Figure 4.3 – Existing object trackers performance decline when objects collide in a moving two object Sound-20K virtual scene whereas AVOT improves with audio onset .....	56
Figure 4.4 – Example AVOT ground truth and audio-visual inputs .....	58
Figure 4.5 – Training and validation loss for SSD– and AVOT .....	59
Figure 4.6 – Comparison of CSRT and SSD to our AVOT method for multi-object tracking .....	61
Figure 4.7 – Examples of AVOT applied to virtual scene of Sound-20K with predicted bounding box .....	62
Figure 5.1 – Audio-augmented rendering of an indoor scene with open and closed reflective surfaces .....	63
Figure 5.2 – Our audio-visual EchoCNN convolutional neural network classifies open or closed surface, depth, and material .....	65
Figure 5.3 – Staged approach to enhance scene and object reconstruction with audio-visual data .....	67
Figure 5.4 – Mel-scaled spectrograms of recorded impulses of different sound sources used .....	70
Figure 5.5 – Sample visualizations of the filters for the two convolutional layers in the audio-based EchoCNN-A neural network .....	73
Figure 5.6 – Spectrograms from a recorded hand clap in front of an interior glass shower door and exterior glass window .....	74

Figure 5.7 – Listener at different distances of 1, 2, 3 ft from sound source in a virtual environment....	75
Figure 5.8 – EchoCNN-A confusion matrices to classify open/closed and depth for an interior glass shower door .....	77
Figure 5.9 – Audio input (i.e. mel-scaled spectrogram) which would produce the highest activation for a given depth class from 1 ft, 2 ft, and 3 ft away from an object .....	78
Figure 5.10 – Evaluation on real and virtual scenes and comparison to depth estimates based on related work .....	80
Figure 5.11 – Echoreconstruction of a TV on a dresser.....	81
Figure 5.12 – EchoCNN may also be used to reconstruct the audio of a scene from video .....	81
Figure 6.1 – Our 3D-MOV neural network is a multimodal LSTM autoencoder optimized for 3D reconstructions of single ShapeNet and multiple Sound20K objects from video .....	84
Figure 6.2 – System overview of object tracking, sound source separation, audio-visual LSTM autoencoder, and 3D decoder for object reconstruction .....	86
Figure 6.3 – We build multimodal datasets using modal sound synthesis for audio and images of voxelized objects as an estimate of shape .....	88
Figure 6.4 – We separately train audio and visual autoencoders to learn encodings and fine-tune for our 3D reconstruction task.....	90
Figure 6.5 – Hidden layer representations are trained to spatially encode with LSTM layers for temporal consistency.....	92
Figure 6.6 – Reconstructed objects from using multiple frames and impact sounds .....	93
Figure 6.7 – 3D-MOV-AV reconstructed image and audio inputs for single view voxelized ModelNet10 classes .....	96
Figure 6.8 – Reconstructed objects from using multiple frames and impact sounds .....	96
Figure 6.9 – Training loss and accuracy for ModelNet10 dataset of 60 epochs .....	97
Figure 6.10 – Training loss and accuracy for Sound20K dataset of 40 epochs using a sequence size of 10. ....	97

## LIST OF ABBREVIATIONS

2D	Two-dimensional
3D	Three-dimensional
3D-MOV	3D Reconstruction of Multiple Objects from Video
ARD	Adaptive Rectangular Decomposition
AR	Augmented Reality
AV	Audio-Visual
AVOT	Audio-Visual Object Tracker
BEM	Boundary Element Method
CAD	Computer-Aided Design
CNN	Convolutional Neural Network
COCO	Common Objects in Context
CRF	Conditional Random Field
CSRT	Channel and Spatial Reliability Tracker
CV	Computer Vision
CW	Continuous Wave
DOF	Degrees of Freedom
FEM	Finite Element Method
FDTD	Finite Difference Time Domain
FOV	Field of View
fps	frames per second
ft	feet
GAN	Generative Adversarial Network
GPU	Graphics Processing Unit
HMD	Head Mounted Display
HMM	Hidden Markov Model
Hz	Hertz

ICP	Iterative Closest Point
IMU	Inertial Measurement Unit
IOU	Intersection over Union
IR	Infrared
LSTM	Long Short-Term Memory
MFB	Multi-modal Factorized Bilinear
MN	ModelNet
MSE	Mean Squared Error
MVS	Multi-View Stereo
NA	Numerical Acoustics
NLP	Natural Language Processing
NMF	Non-Negative Matrix Factorization
NMS	Non-Maximum Suppression
PPSU	Polyphenylsulfone
PSNN	Pouring Sequence Neural Network
PSR	Poisson Surface Reconstruction
PW	Pulsed Wave
RGB	Red, Green, and Blue
RGBD	Red, Green, Blue, and Depth
RNN	Recurrent Neural Network
SfM	Structure-from-Motion
SfS	Shape-from-Shading
SLAM	Simultaneous Localization and Mapping
SPME	Single-Point Multipole Expansion
SSD	Single Shot MultiBox Detector
SSS	Sound Source Separation
TCN	Temporary Convolutional Networks
USB	Universal Serial Bus
VOC	Visual Object Classes
VR	Virtual Reality

## CHAPTER 1: INTRODUCTION

Sights and sound are all around us, in both real and virtual worlds. At times, it is useful to unmute our speakers or put on earphones so we can not only see what looks real but also hear how it sounds. By modeling how objects and fluids should behave and sound, we can generate impact sounds of a user striking an object, colliding sounds of objects interacting with one another, and other object and environment sounds. While visual information provides a great amount of context for what one can expect, auditory cues can assist with complementary data; for example, by differentiating between visually similar materials or the type and amount of liquid in an opaque container. Audio can also provide primary data when vision is unavailable; for example, unlit scenes or sounds outside the current field of view. Whether modeling by physically based phenomena or training a multimodal neural network, both audio and visual information play an important role in learning and processing of data for scene and object understanding.

Not only can additional modes of data provide more information, they can also reaffirm one another. For example, if we only read the captions of a video, we may miss context of the scene or speakers. Information from auditory cues may be lost. Furthermore, we can also use the sound to verify the visual and textual data based on the synchronization between modes. We learn this through experience and understanding. The uncanny valley effect is also described in terms of visual resemblance; however, the same can be applied to our perception of sound, though likely not as pronounced. The effect of audio and visual together matters as well. Individual modes and their interplay represent research areas in multimodal learning, cross-modal self supervision, and transfer learning, to name a few. This dissertation contributes to the simulation of sound for rigid body objects, with and without liquid, and uses both real and synthetic audio as an additional mode to visual data for learning and processing tasks.

### 1.1 Motivation

*‘ The screen is a window through which one sees a virtual world. The challenge is to make that world look real, act real, sound real, feel real. ’*  
(Sutherland, 1965)



**Virtual environments:** as the excitement and economic potential for interactive virtual reality (VR) and augmented reality (AR) increases, modeling of the physical world is imperative to realism of an immersive experience, a sense of being there. In addition to sight, sound is also integral to the level of immersion and sense of presence in virtual and augmented reality (Cummings and Bailenson, 2015). This interest has motivated prior work in 3D sound synthesis, particularly in real-time. Audio has been used to guide user attention and highlight parts of a scene outside current field of view. In the case of redirected walking, sound can also serve as a distraction. The virtual scene can be manipulated in such a way that the user can travel through a virtual world that is larger than the physical working space without the user noticing. Whether or not audio and visual information are processed together in a single pipeline or separately, the presentation of the modes should be synergistic to prevent a distraction from any of the senses which can cause a 'break in presence' (Sanchez-Vives and Slater, 2005).

**Sound synthesis:** traditionally, sounds have been added post-production by Foley artists who recreate sounds for film and other media. However, today's virtual environments expect real-time interactivity. Therefore, game engines and VR systems are incorporating physically-based graphics and sound simulation algorithms for interactive and realistic effects to help users remain immersed in the experience. This real-time modeling of a user interacting with objects in a virtual environment can be done based on the vibration and deformation of an object when it is struck or colliding with another surface. The surrounding air rapidly compresses (compression) and expands (rarefaction or decompression) as the object vibrates outward and inward respectively, As it oscillates periodically, pressure waves are created and air pressure amplitude changes up and down over time. This periodic pattern of compression and rarefaction is known as harmonic motion. Although we may not see the vibrations or deformations, our ears hear the variation in air pressure as sound. This harmonic motion can be modeled as an underdamped spring mass system.

$$mx'' + dx' + kx = 0 \quad (1.1)$$

where  $m$  is mass,  $d$  is damping,  $k$  is stiffness, and  $x$  is displacement. For a volumetric object, the system can be modeled by the sound dynamics equation:

$$Mu'' + Du' + Ku = f \quad (1.2)$$

where  $M$  is the mass matrix, where mass is located on the object,  $u$  is the displacement of each element,  $D$  is the viscous damping matrix (i.e. how velocity decays over time),  $K$  is the stiffness matrix (i.e. defining the connectivity of the elements) and  $f$  is the vector of forces (i.e. inducing vibrations). Note that upper case denotes matrices, lower case denotes vectors or scalars, and  $M$ ,  $D$ , and  $K$  are size  $3n \times 3n$  sparse matrices for  $n$  tetrahedral mesh nodes. Stiffness is based on the objects mesh and Poisson's ratio; mass, construction method Consistent Mass Matrix (CMM); and damping, Rayleigh damping (also known as linearly proportional damping).

$$D = \alpha_1 * M + \alpha_2 * K \quad (1.3)$$

where  $\alpha_1$  and  $\alpha_2$  are real-value parameters. Given these parameters, we can simulate the vibration of the solid volume body object in response to an impulse.

Sound synthesis and physically-based sound synthesis for rigid bodies as well as liquids have been previously studied. A few other major categories include fractures, fire, and thin shell. Since both sound and graphics can be physics-based, the graphics pipeline can be naturally extended to generate sound. Humans can hear frequencies from between 20 Hz to about 22 kHz, requiring applications to sample at a rate of 44 kHz based on the Nyquist Theorem, doubling the frequency we can sense. Rigid body sound has been modeled using modal analysis to decouple the problem into  $n$  independent, damped vibration equations. By performing modal analysis to precompute frequency and damping for a given object and material, real-time sound synthesis can be achieved (O'Brien et al., 2002; Ren et al., 2013a; van den Doel et al., 2001). This is important such that the audio and visual information rendered from interactions of virtual objects with other objects, liquids, and the user reflect the current state of the virtual environment. At runtime, sounds are dynamically created with modal synthesis based on hit (or contact) point where the object is struck and impulse direction. Typically, we simulate an impulse direction normal to the contact point but could synthesize tangentially to the object like  $(0,1,0)$  for  $(x,y,z)$  impulse direction. The solution to the sound dynamics equation is damped sinusoidal waves.

$$q_i = a_i * e^{-d_i * t} \sin(2\pi * f_i * t + \theta_i) \quad (1.4)$$

$$d_i = \frac{1}{2}(\alpha_1 + \alpha_2 * \lambda_i) \quad (1.5)$$

$$f_i = \frac{1}{2\pi} \sqrt{\lambda_i - \left(\frac{\alpha_1 + \alpha_2 * \lambda_i}{2}\right)^2} \quad (1.6)$$

where  $q_i$  is the displacement,  $a_i$  depends on run-time impulse, and  $d_i$  along with  $f_i$  depend on geometry and material properties. Precomputing features, clustering sources, decoupling equations, and simplifying the computational model are techniques that have been used to achieve real-time performance and are contributions to this dissertation.

**Fluid-structure interactions:** solid-fluid interfaces may be referred to as elasto-acoustic coupling. It involves the interactions between the vibrations of an elastic structure and the sound field in the surrounding fluid. However, previous sound synthesis research focuses on single systems only, either solid or liquid but not both. Furthermore, the coupling of systems needs to meet no-penetration and no-slip conditions.

$$\left(\frac{\partial u}{\partial t} - v\right) \cdot n = 0 \quad (1.7)$$

The no-penetration condition (Equation 1.1) occurs at the fluid-structure boundary where  $u$  is the deformation of the solid,  $v$  is the velocity of the fluid, and  $n$  is the outward normal.

$$\left(\frac{\partial u}{\partial t} - v\right) \times n = 0 \quad (1.8)$$

The no-slip condition (Equation 1.1) holds that the tangential velocity components have to be equal. If both independent boundary conditions hold, we have  $\frac{\partial u}{\partial t} = v$ .

While the dynamic sound model may be generalized to any object represented by a tetrahedral mesh and fluid by Lagrangian particles, it needed to be extended to meet these conditions and account for the coupling of these objects that contained a liquid and in real-time. Each mode can perform in real-time separately but the coupling of fluid-structure interactions can be compute intensive, so the problem was simplified to a single system by assuming:

1. Solid object is impermeable
2. Fluid is incompressible

3. Fluid motion coincides with structure motion (also known as a non-moving domain)
4. Fluid is at rest in hydrostatic equilibrium
5. Fluid is inviscid

By assuming that the structural vibration must move the liquid along with the structure, the weight of the surrounding liquid can be added to the system as an added mass effect by modifying the mass matrix of the structure object. This is referred to as a rigid double body where the added mass is the additional drag force resulting from fluid acting on a structure. In reality, the fluid will be accelerated but for simplicity, liquid is modeled as a volume moving with the object.

$$Mu'' + Du' + Ku = f(t) - m_a * u'' \quad (1.9)$$

The resting forces along the boundary of the fluid-structure interface are calculated and included as an added mass to the sound dynamics equation, extending the sound synthesis pipeline to objects containing a liquid. Using an added mass operator simplified a coupled problem into a single fluid-structure system.

**Multimodal learning:** explores relationships between different modalities. Many learning-based methods often primarily rely on visual feedback and human interaction; for example, state-of-the-art vision-based techniques for image classification (Deng et al., 2009) and object detection (Liu et al., 2016; Redmon et al., 2016; Ren et al., 2015a) in images and video, to name a few. Prior work for the liquid pouring task in robotics have also used visual sensing for volume estimation and tracking. With many of these methods using video as an input, sound may readily be available for multimodal learning with both audio and visual data which can improve processing and performance. Fused modalities also cover edge cases that can be a challenge for a single model such as noise from blur, poor illumination, or occlusions can cause error for visual data. On the other hand, environmental noise, varying room acoustics, or mixed audio from other sound sources can prove difficult with only audio inputs.

Various techniques have been used to fuse multiple modalities of data. Natural Language Processing (NLP) has demonstrated the use of multimodal learning for visual question and answering systems (Fukui et al., 2016; Ilievski and Feng, 2017), video captioning (Pasunuru and Bansal, 2017; Wang et al., 2018). Audio-visual have also been used for speech separation (Zhao et al., 2018) and object classification (Anusha and Roy, 2015). Rather than using extra sensors such as contact microphones (Clarke et al., 2018) or ther-

mal imaging cameras (Schenck and Fox, 2017), frames of audio and visual data may be used from the recorded video. In addition to merge layers of concatenation, addition, and multiplication to combine input streams, bilinear modeling has also been used to learn multiplicative interactions of differing input types (Gao et al., 2015; Yu et al., 2017b; Park et al., 2016). A simple multi-modal bilinear model is defined as:

$$z_i = x^T W_i y \quad (1.10)$$

where  $x$  and  $y$  are mode features,  $W_i \in R^{m \times n}$  is a projection matrix, and  $z$  is the output bilinear model.

In addition to multimodal learning in NLP and audio-visual learning, it has also been demonstrated to aid in localization, object, and material classification based on impact sounds of meshed geometries. Furthermore, reconstruction methods also benefit from audio for enhanced reconstruction and material inference. While vision-based methods may have difficulty detecting textureless or glass surfaces, echoes of reflecting sounds from a sound source may be used to estimate the depth of these surfaces and inpaint to fill holes in the reconstructed geometry.

**3D reconstruction:** a number of algorithms exist to generate 3D shape from 2D and other sensory information. A common processing pipeline involves capture, point generation, meshing and texturing, and temporal mesh processing. During capture, the setup often includes some level of calibration, preprocessing, bias correction, and background subtraction. Passive methods use sensors (e.g. camera) to capture details (e.g. RGB) about an object for reconstruction without any interference or projections into the scene. On the other hand, active reconstruction techniques (e.g. RGB-D) use infrared projectors to illuminate and detectors to measure the radiance on the object’s surface. Using commodity sensors such as the Microsoft Kinect and GPU hardware allow for both static (Golodetz\* et al., 2015; Izadi et al., 2011) and dynamic (Dai et al., 2017b; Newcombe et al., 2015) scenes to be scanned in real-time. 3D scene reconstructions have also used sound such as time of flight sensing (Crocco et al., 2016). Meshing and texturing may include topology denoising, island removal, hull-constrained PSR, occlusion detection, and texturing

Results of reconstruction of 3D geometries can also serve inputs to other learning based algorithms. For instance, 3D point clouds from depth maps, multimodal MVS, or iterative surface estimation have been used as inputs to train neural networks for other downstream tasks such as object classification, segmentation, and tracking (Qi et al., 2016a). Reconstruction research has generated large amounts of 3D

scene (Silberman et al., 2012a; Song et al., 2017) and object (Lai et al., 2011; Singh et al., 2014; Wu et al., 2015b) data that can be used for training vision-based neural networks for classification, segmentation, and other downstream tasks.

## **1.2 Scope of this dissertation**

There are a number of training datasets, neural network architectures, technologies, and active research areas for multimodal learning, especially in the area of audio and visual data from video. Applications in these areas range from Virtual and Augmented Reality, e.g., sound synthesis, reconstruction, inference, etc. to expanding methods to handle a wider variety of surfaces and scenes, such as illumination, reflectivity, texture, and occlusion. This dissertation focuses on coupling fluid-structure (chapter 2) and audio-visual classification (chapter 3), tracking (chapter 4), and reconstruction (chapter 5 and chapter 6) with demonstrations in multimodal learning and virtual reality.

## **1.3 Thesis Statement**

My thesis statement is as follows:

*Coupling multimodal information enhances task performance and processing of audio-visual learning based methods for fluid-structure sound synthesis, liquid pouring sequences, object tracking, and 3D reconstructions while also allowing for single mode application for special cases.*

To support this thesis, I present four methods; one method to efficiently synthesis sound of objects containing a liquid, two methods to accurately estimate liquid pouring sequences and track objects using audio-visual neural networks, and one method to use audio to enhance scene and object reconstructions using mobile devices.

## **1.4 Main Results**

### **1.4.1 Sound Synthesis for Fluid-Structure Coupling**

Previous sound synthesis research has focused on single systems only, either solid or liquid but not both. Since not all single mode, sound simulations achieve real-time performance, modeling the variation in sound from a coupled vibrating fluid-structure system could be computationally expensive. This work

was the first to synthesize sound for a system containing both a rigid body object and liquid (referred to as a fluid-structure coupling).

In chapter 2, I present a fast and practical method for simulating the sound of rigid body objects that contain liquid. To maintain real-time, interactive performance, we modify the existing modal synthesis pipeline by adding pre-processing steps. Those steps are to identify mesh nodes of the object that bound the liquid and to then modify the mass matrix of the structural object by an amount proportional to the liquid density and volume.

The main contributions of my work are:

1. Transforming the problem into a single fluid-structure system using the *added mass operator*;
2. Enhancing the sound synthesis pipeline with pre-processing steps for objects containing a liquid;
3. demonstrating the proposed method in interactive 3D VR applications.

Actual recordings versus synthesized frequencies were compared for varying amounts of liquid and results were less than 5% relative error. The interactivity of the algorithm was demonstrated with VR applications of a simulated liquid xylophone and kitchen scene of different containers, liquids, and volumes.

### **1.4.2 Analyzing Liquid Pouring Sequences via Audio-Visual Neural Networks**

Prior work to estimate liquid poured amounts often require predefined amounts in the source container or rely on visual data. To compensate for vision-based challenges such as occlusion and transparency, this work uses audio from the pouring sequence to augment audio and visual only methods.

In chapter 3, I introduce audio and audio-visual neural networks in the form of multimodal convolutional neural networks (CNNs) to perform weight estimation, overflow detection, and content and container classification for robots pouring liquids.

The main contributions are:

1. Training, validation, and test data generated from audio recordings and video images with ground truth measurements from a digital scale
2. Audio-based CNN for multiclass weight estimation and binary classification for overflow detection by robotic systems

3. Audio-augmented neural network enhancing the audio only based method with fused visual inputs for robots pouring contents into various target containers
4. Pouring content and target container classification for robots, based on pouring sequence audio data

Upto 91.5% of the audio intervals for the robot pouring sequences were classified within 0.4 oz using audio-visual data. This resulted in an average error of 0.2 oz. The sound from pouring the liquid was also used to predict the type of liquid and target container.

### **1.4.3 Audio-Visual Object Tracking of Multiple Objects**

Visually based object trackers can run into challenges when object collide, occlude, or appear similar but differ in material. By using audio of the impact sounds from object collisions, rolling, etc., an audio-based technique may be used in conjunction with other neural networks to augment visually based object detection and tracking methods. In chapter 4, I describe the first use of an audio-visual neural network for tracking tabletop sized objects and enhancing visual object trackers.

The key contributions of this work include:

1. An end-to-end, jointly trained audio-visual object tracker (AVOT) to enhance visual object tracking
2. Ground truth bounding boxes for virtual scenes from the Sound-20K audio-visual dataset with 1, 2, and 3 objects
3. Scheduler for object detection re-initialization based on audio onset detection when using multi-modal tracking

By fusing audio with visual data, the audio-visual object tracker (AVOT) achieves upto 78% intersection over union (IoU) post-collision tracking accuracy compared to 69% using state-of-the-art deep learning visual trackers.

### **1.4.4 Audio-Augmented Scene and Object Reconstruction**

In chapter 5, I introduce echoreconstruction, an audio-visual method that uses reflecting sounds to aid in geometry and audio reconstruction. Scenes containing open and reflective surfaces often lead to existing techniques reconstructing objects behind (in the case of transparent glass) or in front of (in the case of



mirrors) the object. By using pulsed audio from a mobile device, inferences from a convolutional network can detect and estimate depth to the reflecting surface. Key results include:

1. EchoCNN, a fused audio-visual CNN architecture for classifying open/closed surfaces, their depth, and material
2. EchoReconstruction, a staged audio-visual 3D reconstruction pipeline that uses mobile phones to enhance scene geometry containing windows, mirrors, and open surfaces with depth filtering and inpainting based on EchoCNN inferences
3. Semantic rendering of window and mirror in audio-augmented reconstructions based on point of view (e.g. environment outside of the window or reflected view of a TV)
4. Real and synthetic audio-visual ground truth data for multiple scenes containing windows and mirrors in addition to reflection separation data (direct, early, or late reverberations)

Overall, 71.2% of hold out reflecting sounds were correctly classified as an open or closed boundary and 71.8% of 1 second audio frames were correctly classified as 1 ft, 2 ft, or 3 ft away from the surface based on audio alone; 89.5% when fused with its corresponding image. Pulsed sounds were emitted a maximum of 3 feet away to remain in the free field. Beyond that, there will be less noise reduction due to reflecting sounds in the reverberant field (Egan, 1988).

In chapter 6, I detail a multimodal single and multi-frame LSTM autoencoder for 3D reconstruction using audio-visual input. Existing methods may experience difficulties in cluttered environments with multiple objects causing occlusion. To address such limitations, the method adds audio as another input, specifically *impact sounds* resulting from object to object or scene interactions. The main contributions of this work can be summarized as:

1. A multimodal LSTM autoencoder neural network for geometry and material reconstruction from audio and visual data
2. The resulting implementation has been tested on voxel, audio, and image datasets of objects over a range of different geometries and materials
3. Experimental results of our approach demonstrate the reconstruction of single sounding objects and multiple colliding objects in a virtual scene

#### 4. Audio-augmented datasets with ground truth object tracking bounding boxes

Single view ShapeNet resulted in IoU metrics of 21.2% for audio and 32.6% for audio-visual. 10 Sound20K views resulted in 37.15% and 69.8% IoU for audio and audio-visual respectively.

### 1.5 Contributions of this dissertation

For sound synthesis: (1) Transforming the problem into a single fluid-structure system using the *added mass operator*. (2) Enhancing the rigid-body sound synthesis pipeline with pre-processing steps for objects containing a liquid. (3) Demonstrating the proposed method in interactive 3D VR applications.

For analyzing pouring sequences: (1) Training, validation, and test data generated from audio recordings and video images with ground truth measurements from a digital scale. (2) Audio-based convolutional neural network for multi-class weight estimation and binary classification for overflow detection by robotic systems. (3) Audio-augmented neural network enhancing the audio only based method with fused visual inputs for robots pouring contents into various target containers. (4) Pouring content and target container classification for robots, based on pouring sequence audio data.

The broad contributions of this dissertation are new real-time fluid-structure coupling methods, new audio-visual classification and tracking, and prototype audio-augmented object and scene reconstruction on mobile devices.

### 1.6 Organization

The remainder of this dissertation is organized as follows. The discussion of fluid-structure coupling begins with the sound synthesis of objects containing a liquid using the added mass operator in chapter 2. This is followed by my work on analyzing liquid pouring sequences using audio-visual neural networks in chapter 3. Next, I cover my method for audio-visual object tracking in chapter 4. Finally, I discuss a model for using audio on mobile devices to enhance 3D reconstructions of scenes (chapter 5) and objects (chapter 6). I conclude my dissertation in chapter 7 by presenting a summary of this work, its contributions, and a discussion of future work.

## CHAPTER 2: SOUND SYNTHESIS FOR FLUID-STRUCTURE COUPLING<sup>1</sup>

This chapter describes a fast and practical method for simulating the sound of non-empty objects containing fluids. Our sound synthesis algorithm for fluid-structure coupling enhances the rigid-body sound with an added mass operator to account for the amount and type of liquid contained within the object. Although the technique assumes the liquids are inviscid, remain steady, and are not mixed and has some limitations (such as granularity of the solid discretization since modifications to the mass matrix occur at mesh nodes), this method is designed and demonstrated for use in interactive 3D systems, where live sound synthesis is important.

### 2.1 Introduction

Sound is integral to the level of immersion and sense of presence in virtual and imaginative environments (Cummings and Bailenson, 2015). Research has demonstrated that even with eyes closed, we can have a mental imagery response similar to a visual perception, as if we are actually viewing the objects (Kosslyn, 2005). However, in the case of virtual reality, a distraction from any of the senses can cause a ‘break in presence’ (Sanchez-Vives and Slater, 2005). Therefore, game engines and VR systems are incorporating physically-based graphics and sound simulation algorithms for interactive and realistic effects to help users remain immersed in the experience.

Given an object and its material properties, we can simulate, in real-time, the resulting sound that it would make when struck (van den Doel et al., 2001; O’Brien et al., 2002; Ren et al., 2013a). To achieve this real-time sound synthesis, we may pre-compute the frequencies and dampings of each element node by performing modal analysis using a generalized eigendecomposition. Then, sound synthesis is based on the results of these pre-processed steps and a given impulse force.

---

<sup>1</sup> Most of this chapter previously appeared as an article in *The Visual Computer*. The original citation is as follows: Justin Wilson, Auston Sterling, Nicholas Rewkowski, and Ming C. Lin. Glass half full: sound synthesis for fluid–structure coupling using added mass operator. *The Visual Computer*, 33(6):1039–1048, Jun 2017. ISSN 1432-2315. doi: 10.1007/s00371-017-1383-8. URL <https://doi.org/10.1007/s00371-017-1383-8>

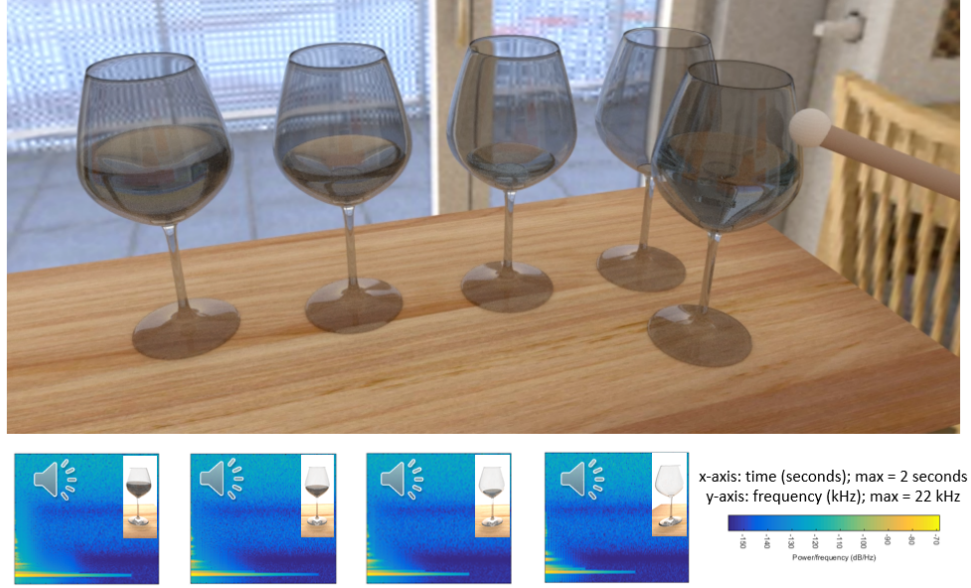


Figure 2.1: (Top) Simulated liquid xylophone with varying levels of liquid and/or density based on our sound synthesis algorithm for fluid-structure coupling that enhances the rigid-body sound modeled by accounting for both the amount and type of liquid contained within the object. (Bottom) Spectrograms for each goblet, illustrating how the fundamental frequency decreases, resulting in a lower pitch as the amount of liquid or density increases

Since both sound and graphics are physics-based, the rendering pipeline can be naturally extended to generate sound based on the same physics. However, 3D objects for graphical display can be filled with liquids (and/or other materials) in the scene and we can hear sounds coming from sources that we cannot see. It is, therefore, important to simulate sounds from non-empty objects that may be hollow by design but contain liquid in the virtual world.

In this paper, we present a new and practical method that satisfies both audio and graphics requirements and enhances the existing physically-based sound synthesis model to include non-empty objects. This feature is important, because we interact with filled containers and can distinguish between objects that are either full or empty (Rocchesso et al., 2003). Therefore, the difference in perceived sounds between empty and non-empty containers should be modeled to increase the realism of the virtual environments.

### 2.1.1 Contributions

This chapter’s main contributions are:



Figure 2.2: A wooden pot, metallic teapot, and porcelain bowl with fine and coarse subdivision surfaces are a few of the objects simulated with varying volumes of liquid (e.g. water). The far right image is the same porcelain bowl but simulated with a more dense liquid (e.g. milk)

1. Transforming the problem into a single fluid-structure system using the *added mass operator*;
2. Enhancing the rigid-body sound synthesis pipeline with pre-processing steps for objects containing a liquid;
3. Demonstrating the proposed method in interactive 3D VR applications.

## 2.2 Related Work

**Sound Synthesis:** Research has modeled rigid body sound using modal analysis to decouple the sound dynamic equations into  $n$  independent, damped vibration equations (Adrien, 1991; van den Doel and Pai, 1998). These sound synthesis techniques create sound based on vibration analysis of the object resulting in varying frequency vibration modes. Modal analysis relies on expensive pre-processing to achieve interactive runtime performance. In addition to rigid-body sounds, there are a few other major physically based categories such as fracture (Zheng and James, 2010), fire (Chadwick and James, 2011), and liquids (Moss et al., 2010), to name a few. Please see a recent survey (James, 2016) for more details.

**Parameter Acquisition:** To perform sound synthesis, object specific parameters are required, for example material specific damping coefficients that have traditionally been tuned manually. To automatically determine these material properties, a method to extract parameters from recorded audio was introduced (Ren et al., 2013a). Alternative and more general damping models have also been introduced (Sterling and Lin, 2016).

**Acoustic Transfer:** However, what we hear is not the modal amplitudes of the vibrating solid but the sound pressure waves radiating from the vibrating surface into the surrounding air. Acoustic transfer techniques couple synthesis and propagation together to generate sound. There are geometric acoustics (GA) techniques (Funkhouser et al., 2003) and numerical acoustics (NA) methods that solve the wave equation using adaptive Finite Element Method (FEM) (Thompson, 2006), Boundary Element Method (BEM) (Brebbia and Ciskowski, 1991), Finite Difference Time Domain (FDTD) (Sakamoto et al., 2006), spectral methods (Boyd, 2001), and Adaptive Rectangular Decomposition (ARD) (Raghuvanshi et al., 2009).

**Coupled Synthesis-Propagation:** In addition to focusing on outgoing waves from the vibrating surface into the air, research also been conducted to evaluate cavity tones for virtual instruments (Ren et al., 2012) and aerodynamic sound of a swinging sword (Dobashi et al., 2003). To simulate the sound propagation into the full 3D environment, various methods have been developed using geometric sound propagation based on ray tracing techniques (Rungta et al., 2016a), wave-based algorithms (Mehra et al., 2015b), and two types of multipole expansion for radiating sound fields-multipole expansion based on equivalent source methods (James et al., 2006a) and single point multipole expansion (Rungta et al., 2016a). For single-point multipole expansion (SPME), a single multipole source is placed inside the object while multi-point multipole expansion places a large number at different points inside the object. These representations depict outgoing pressure fields. Source clustering has also been proposed to reduce computation since complexity varies with the number of sound sources (Tsingos et al., 2004a).

**Fluid-Structure Mechanics:** To allow for interactions between a solid and a liquid, Müller et al. (Müller et al., 2002, 2004) simulated the interaction of fluids with deformable solids, which estimates the forces between virtual boundary particles of the solid surface and fluid particles. Computational approaches are discussed by Bazilevs et al. (Bazilevs et al., 2013), where the variational structural mechanics equation in matrix form may be written as:

$$M\ddot{Y} + D\dot{Y} + KY = F \quad (2.1)$$

$$M_{AB} = \int_{\Omega} N_A N_B d\Omega \quad (2.2)$$

where  $M = M_{AB}$ . For linear elastodynamics, a density term is added to the mass matrix (and damping matrix as well since it is a linear combination of the mass and stiffness matrices). Linear elastodynamics:

$$\mathbf{M} = [\mathbf{M}_{ij}^{AB}]$$

$$M_{ij}^{AB} = \int_{\Omega} N_A \rho N_B d\Omega \delta_{ij} \quad (2.3)$$

Here,  $\Omega$  is the material domain of a structure along with the boundary  $\Gamma$ ,  $\rho$  is the mass density of the structure, and  $N$  is associated to its unique mesh nodes.

**Added Mass Operator:** Viewing the coupled fluid and structure as a single system, added mass or virtual mass is a concept in fluid mechanics that incorporates the surrounding fluid of an accelerating or decelerating object (Newman, 1977). In the marine industry, added mass is often referred to as hydrodynamic added mass and can reach up to 1/3 of the total ship mass. Although less common in aeronautics because of small air density (except lighter-than-air balloons or blimps), the stability and convergence properties were evaluated based on the ratio of added-mass to the actual structural mass (Brummelen, 2009). More information on the topic of added mass and fluid inertial forces can be found in a survey by the the Naval Civil Engineering Laboratory (Brennen, 1982).

## 2.3 System Overview

In this paper, we introduce an enhanced sound synthesis model that accounts for the amount of liquid contained within an object by incorporating these added mass and fluid-structure interactions into the existing rigid body sound synthesis model. Our new approach for simulating sound of non-empty objects begins by modifying the mass matrix along the fluid-structure interface prior to performing modal analysis on the system. We present various approaches to calculate the modification required for a specific liquid density and volume.

### 2.3.1 Rigid-Body Sound Synthesis Pipeline

1. Generate the object's volumetric mesh and stiffness  $\mathbf{K}$ , mass  $\mathbf{M}$ , and damping  $\mathbf{D}$  matrices
2. Apply an added mass operator to modify the mass matrix based on a fluid-structure interface
3. Run generalized eigen-decomposition
4. Construct decoupled modal vibration equations
5. Apply an impulse force

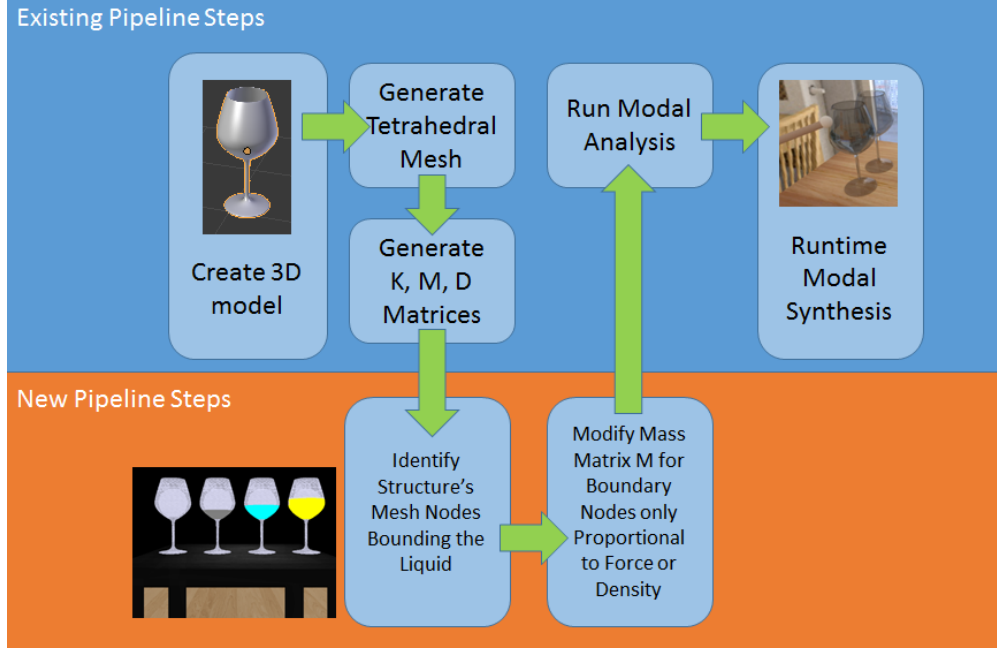


Figure 2.3: Overview of the enhanced rigid body sound pipeline to include new contributed steps to account for different liquid volumes and densities contained within the solid object. These new pre-processing steps come before modal analysis and therefore can be computed before simulation allowing for usage in real-time interactive systems

## 6. Numerically integrate individual modes

### 2.3.2 Modal Sound Analysis

Modal analysis is the standard linear model for dynamic deformation and physically based sound (Shabana, 1997). When an object is struck, it vibrates and deforms. The surrounding air rapidly compresses (compression) and expands (rarefaction or decompression) as the object vibrates outward and inward respectively. As it oscillates periodically, pressure waves are created and air pressure amplitude changes up and down over time (like a sinusoidal wave). Although we may not see the vibrations or deformations, our ears hear the harmonic oscillation and periodic pattern of compression and rarefaction in air pressure as sound. We can simulate the vibration of the solid volume body object in an underdamped response to an impulse using the following equation:

$$M\ddot{u} + D\dot{u} + Ku = f \quad (2.4)$$



where  $M$ ,  $D$ , and  $K$  are the mass, damping, and stiffness matrices, respectively;  $u$  is the displacement vector and  $f$  as the force vector. It is well-established to approximate small levels of damping with *Rayleigh Damping*, i.e. representing the damping matrix as a linear combination of the mass and stiffness matrices.

### 2.3.3 Modal Sound Synthesis

We can then simulate sound based on a contact position  $p = (x, y, z)$ , where the object is struck. The impulse direction is usually normal to the contact point but could be tangential to the object. To achieve real-time performance, pre-processing steps are performed for a given object and material.

After solving the generalized eigenvalue problem of Eqn. 2.4, the solution are *modes*, i.e. damped sinusoidal waves where each mode has the form

$$q_i = a_i e^{-d_i t} \sin(2\pi f_i t + \theta_i), \quad (2.5)$$

where  $f_i$  is the frequency of the mode,  $d_i$  is the damping coefficient,  $a_i$  is the excited amplitude, and  $\theta_i$  is the initial phase. And,

$$\omega_i = 2\pi f_i = \frac{\sqrt{4km - d^2}}{2m} \quad (2.6)$$

We ignore  $\theta_i$  and safely assume it to be zero since the object is initially at rest. It is also important to note that our approach requires the mass to be normalized ( $m = 1$ ).

### 2.3.4 Added Mass Operator

In solid and structural mechanics and civil engineering where pressure forces are commonly applied to the structure, the additional force resulting from fluid acting on a structure when formulating the system equation of motion is known as *added mass*. In a physical sense, the added mass of an incompressible flow is the weight added to the system from surrounding fluid that the accelerating or decelerating structural vibration must move with the structure. This force is factored into our sound dynamics equation as:

$$M\ddot{u} + D\dot{u} + Ku = f - m_a\ddot{u} \quad (2.7)$$

where  $m_a$  is the added mass. Reordering terms, we arrive at:

$$(M + m_a)\ddot{u} + D\dot{u} + Ku = f \quad (2.8)$$

Note that if mass is also included in the damping matrix, the added mass would automatically be included there as well.

On short time scales, the effect of the fluid on the structure can be represented as an added mass. Ratio of this added mass to the structural mass is critical to convergence and should be less than or equal to 1; else, the system may become unstable (Brummelen, 2009). In reality, the fluid will be accelerated but for simplicity, it is modeled as a volume moving with the object as a second-order tensor, relating the fluid acceleration vector to the resulting force vector on the body.

Added mass is analogous to the amount of work needed to change the kinetic energy  $T$  associated with the motion of the fluid:

$$T = \frac{\rho}{2}IU^2 = \frac{\rho}{2} \int_V (u_1^2 + u_2^2 + u_3^2) dV \quad (2.9)$$

where  $u$  is the fluid velocity in Cartesian coordinates,  $V$  is the volume of fluid,  $\rho$  is the liquid density,  $U$  is the structure's velocity, and  $I$  is a number proportional to the liquid's volume. We assume a non-moving domain and a rectilinear velocity, which could be generalized to other motions if required (?). When the solid body accelerates causing changes in  $U$ , kinetic energy  $T$  increases, supplying additional work by the body on the fluid. The rate of additional work is the rate of change of  $T$  with respect to time  $dT/dt$  and is considered added drag by the body. A viscous fluid also contributes a drag force; however, in our implementation, we assume inviscid liquids. Then, the added drag,  $F$ , can be represented as:

$$F = -\frac{1}{U} \frac{dT}{dt} = -\rho I \frac{dU}{dt} = -\rho \frac{V}{2} \ddot{u} \quad (2.10)$$

$$I = \int_V \frac{u_i u_i}{U U} dV \quad (2.11)$$

where  $I$  is half the volume and  $F$  is in the form of  $m_a \ddot{u}$  or  $\rho_{\text{fluid}} \cdot \frac{V_{\text{fluid}}}{2} \ddot{u}$  if we approximate our liquids to be spherical, similar to the concept of spherical bounding volume hierarchy (Gottschalk et al., 1996). Since cylindrical objects result in a full volume rather than half (Brennen, 1982), future work is required

to analyze the trade-offs between ease of implementation and accuracy in accounting for the geometric complexity of the liquid.

As a result, if we approximate our liquid as a sphere, added mass for each liquid object is approximated by:

$$m_a = \rho_{fluid} \cdot V_{fluid}/2 \quad (2.12)$$

For example, the added mass for a sphere (of radius  $r$ ) is  $\frac{2}{3}\pi r^3 \rho_{fluid}$  which is half the volume of a sphere times fluid density. Therefore, a spherical air bubble rising in water has a mass of  $\frac{4}{3}\pi r^3 \rho_{air}$  and added mass of  $\frac{2}{3}\pi r^3 \rho_{water}$ .

Since the liquid must move with the same phase as the structure's motion, this may be referred to as a rigid *double body* (Newman, 1977). In general, the value of the added mass also depends on the direction of acceleration and is incorporated by projecting the area of the body in the direction of acceleration; for instance, tangential acceleration yields zero added mass. In our case, projection is not necessary since we assume a non-moving domain where the fluid motion is in the same direction as the structural vibration.

## 2.4 Sound Synthesis for Fluid-Structure Coupling

Added mass from the liquid is distributed to mesh nodes of the structural object along the fluid-structure boundary. In our system, we represent both the solid and liquids as tetrahedral meshes to detect boundary elements although particle-based methods (such as Lagrangian particles) are also possible. We also make the following assumptions:

- Solid is impermeable & the liquid is incompressible
- Fluid is at rest in hydrostatic equilibrium or the flow velocity at each point is constant over time
- Domain is non-moving, i.e. fluid motion coincides with structure motion
- Fluid is inviscid, having no or negligible viscosity

### 2.4.1 Identifying Nodes at Fluid-Structure Interface

A number of collision detection methods exist (Gottschalk and Lin, 1998) and may be used to identify the necessary boundary nodes. Since the liquid is assumed to be at rest, boundary detection of symmetric

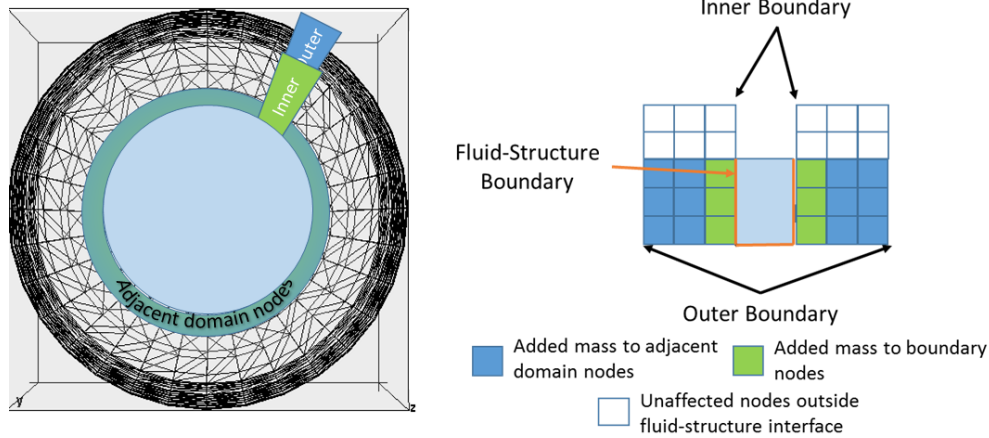


Figure 2.4: Left: top view. Right: cross sectional view. Surface boundary and adjacent domain nodes in between the inner and outer boundary are detected, stored, and modified by an added mass based on the amount and type of liquid in each respective object

objects may be simplified by selecting all nodes above and below the min and max nodes that intersect between the structure and fluid (see Fig. 2.4).

## 2.4.2 Modifying the Mass Matrix

We use the added mass equation for a sphere (see Eqn. 2.12) to calculate total added mass for each liquid based on the liquid density and volume. For example, if the liquid added is pure water, then the density is  $1,000 \text{ kg/m}^3$ ; milk,  $1,050 \text{ kg/m}^3$ ; olive oil, about  $860 \text{ kg/m}^3$ ; and boiling water, approximately  $958 \text{ kg/m}^3$ . This total added mass is then distributed to the mesh nodes of the structural object. An approach to randomly distribute this added mass across the entire structure is less accurate than uniformly or force weighted distributions along the fluid-structure boundary.

A uniform distribution calculates the total added mass and then distributes an equal amount to each boundary and adjacent domain mesh node, as illustrated in Fig. 2.4 and Fig. 2.5. Alternatively, rather than uniformly distributing additional mass from the liquid, we may distribute it relative to the local force. For example, mesh nodes with a greater depth will have a greater force and therefore should obtain a greater proportion of the total added weight. Given the boundary nodes and the weight of the liquid, we modify the mass matrix elements for each node based on its hydrostatic force, as shown in Fig. 2.6 and Eqn. (2.18).

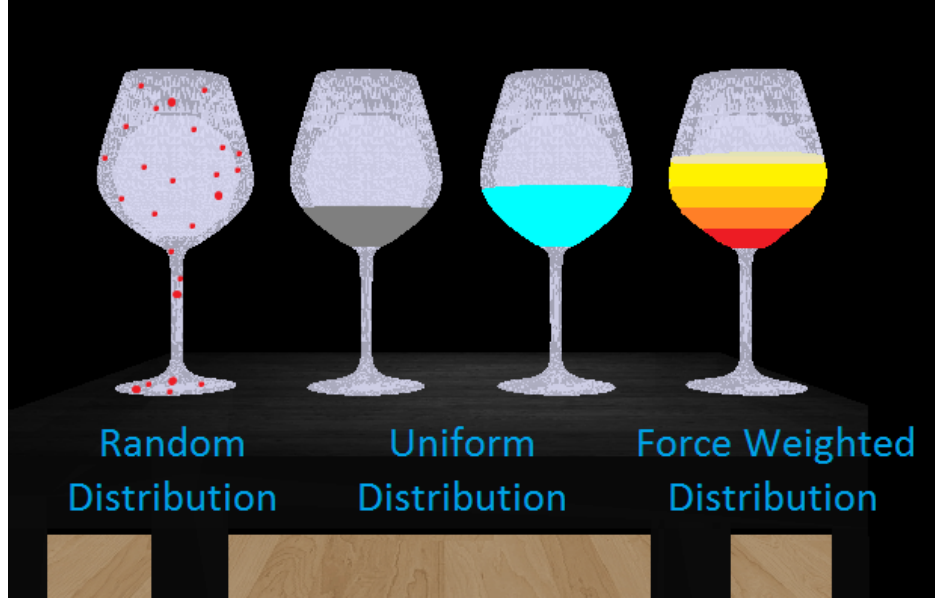


Figure 2.5: Added mass should be distributed along the fluid-structure interface. Randomly distributing (left) the added mass over the entire structure also modifies the sound but uniformly (middle) or weighted (right) distributions result in frequencies closest to real-world recordings

### 2.4.3 Hydrostatic Force on a Curved Surface

In our system, we can also evaluate the force along the boundary of an object to determine the weighted distribution of added mass. This is based on fluid statics where the pressure increases linearly with depth below the water's surface for fluids in a non-moving domain (CAL POLY POMONA, 2016). For example, dams are designed to be parabolic for improved stability, allowing the weight of the water to press the dam into the ground.

Newton's 3rd law dictates that  $\vec{F}_N = -\vec{F}_R$  and 2nd law that  $\sum \vec{F} = m\vec{a} = 0$  since there is no motion. Given these conditions, our goal is to solve for the force along the fluid-structure boundary  $\vec{F}_N = F_H\hat{i} + F_V\hat{j}$  where N stands for normal, R for reactive, H for horizontal, and V for vertical, as illustrated in Fig. 2.6. For both the horizontal and vertical components of the normal force, the force magnitude increases with depth. Therefore, the force along the boundary of the solid is greatest at points farthest away from the surface. In addition to depth, the hydrostatic force equations also demonstrate the linear relationship with density. Since the fluid body is not moving, the net force is zero which means that the

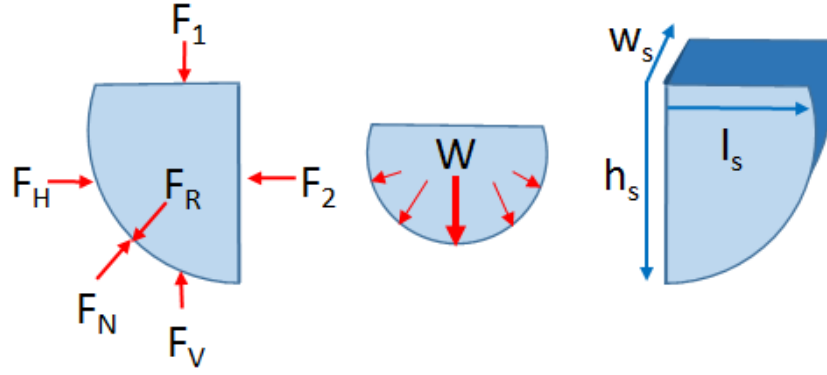


Figure 2.6: Free-body diagram for the hydrostatic force along the fluid-structure boundary of a curved surface depends on depth below the surface, liquid density, liquid volume, and gravitational constant  $g$

horizontal force from the structure is equal to the horizontal pressure force from the liquid.

$$\sum F_x = 0 = F_2 - F_H \quad (2.13)$$

$$F_H = F_2 = \gamma h_c A_{\text{left}} = \gamma \left( d + \frac{h_s}{2} \right) (w_s h_s) \quad (2.14)$$

where  $\gamma = \rho \cdot g$ ,  $h_c$  is the vertical distance from the free surface to the centroid of the left planar surface, area of the left planar surface is  $A_{\text{left}}$ ,  $d$  is depth,  $h_s$  is height from bottom to top of the liquid surface, and  $w_s$  is the width of the liquid surface.

$$\sum F_y = 0 = F_V - F_1 - W \quad (2.15)$$

$$W = m_f g = (\rho V_f) g = \gamma V_f = \gamma (d w_s l_s) \quad (2.16)$$

$$F_V = F_1 + W \quad (2.17)$$

The sum of the forces in the vertical direction are also equal to zero.  $F_1$  is the pressure force due to the weight  $W$  of the fluid directly above the isolated fluid body and in our case is equal to zero. The weight  $W$  is mass  $m$  times the gravitational constant  $g$ , where mass can be calculated as density  $\rho$  times volume  $V$ .

$$F_N = \sqrt{F_H^2 + F_V^2} = F_R \quad (2.18)$$

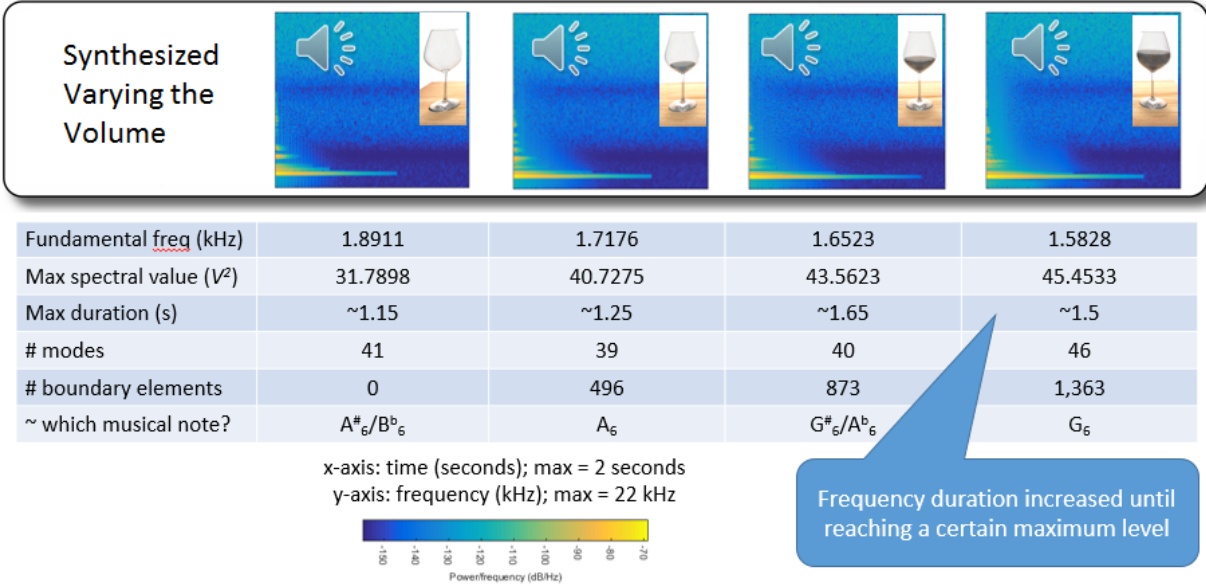


Figure 2.7: This figure displays the results of sound synthesis for our approach with liquid volume increasing from empty to glass half full; notice that the fundamental frequency decreases resulting in a lower pitch as the liquid increases

With this approach, we are able to calculate the forces along the boundary. Rather than uniformly distributing additional mass from the liquid, we can distribute relative to the local force. Given the boundary nodes and the weight of the liquid, we modify the mass matrix elements for each node based on its force contribution. If we calculate the forces using a particle approach, then we can distribute the liquid particle forces to the structure mesh nodes using Gaussian quadrature rules (Müller et al., 2004).

## 2.5 Results and Analysis

We have implemented our algorithm in C++, while performing the power spectral density and spectrogram analysis using MATLAB. The virtual reality demo application was created with Unreal Game Engine and viewed with the HTC Vive, as shown in the supplementary video posted at: <http://gamma.cs.unc.edu/GlassHalfFull>

### 2.5.1 Comparison: Synthesized vs. Recording

To evaluate the effectiveness of our method, we compared synthesized sounds to real-world recordings. The results show that the direction of the change in sound is accurate. For example, the pitch decreases as the volume or density increases. This inverse relationship is referred to as “wet” natural frequencies (with

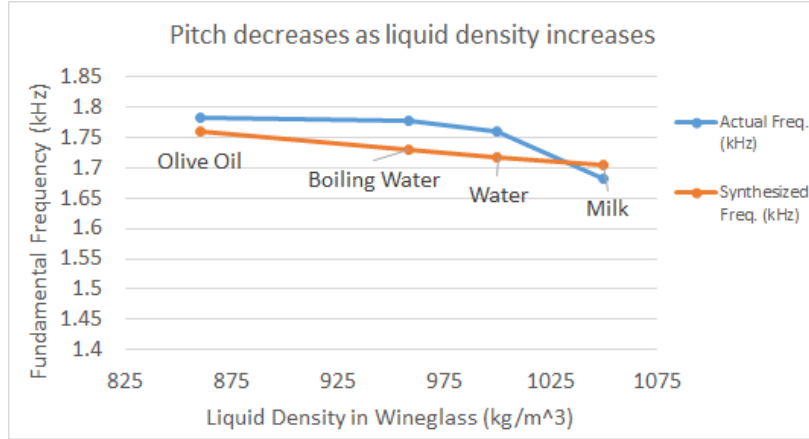


Figure 2.8: Given equal volumes of liquid, the fundamental frequency of the sounding object decreases as the contained liquid density increases

liquid) that are lower than corresponding “dry” natural frequencies (without liquid) (Basic, 2012). Also, although the synthesized and actual frequencies differ, the similarity in sound between the two are imperceptible (Sek and Moore, 2016). Our synthesized sounds would provide the expected auditory difference among different liquids, as shown in Table 2.1.

Keeping the volume constant and changing density (e.g. milk versus water) also changes the frequency. For example, in Figure 2.8, we can see that frequency is inversely proportional to liquid density; that is, as density increases, frequency decreases. The difference in sound by varying the type may be harder to distinguish unless there is a more significant difference in density (e.g. olive oil versus milk).

Table 2.1: Generalizations for different liquid densities. Syn Freq and Actual Freq are the synthesized and actual fundamental frequencies respectively, in kHz, of a wineglass with 1/4 liquid volume

Liquid	Syn Freq (kHz)	Act Freq (kHz)	Rel Error
Milk	1.7037	1.6823	1.27%
Water	1.7176	1.7607	2.45%
Hot Water	1.7297	1.7771	2.67%
Olive Oil	1.7597	1.7824	1.28%

Similar to density, volume is also inversely proportional to pitch. Compared to real-world recordings, the change in frequency direction is aligned with expectations and the magnitude of the frequencies are reasonable within single-digits, (see Fig. 2.9 and Table 2.2).



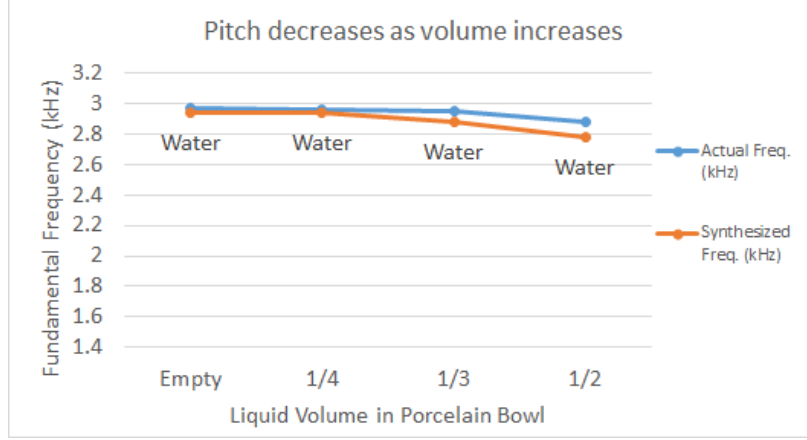


Figure 2.9: Fundamental frequency of the object decreases as the amount of liquid increases

Table 2.2: Generalizations for different liquid volumes where Syn Freq and Actual Freq are the synthesized and actual fundamental frequencies, in kHz, of a porcelain bowl with water

Vol	Syn Freq(kHz)	Act Freq(kHz)	Rel Error
Empty	2.9419	2.9709	0.98%
1/4	2.9389	2.9597	0.70%
1/3	2.8816	2.9453	2.16%
1/2	2.7810	2.8759	3.30%

## 2.5.2 Spectrogram Analysis

We used power spectrograms to analyze the synthesized sounds in a time-varying frequency representation. This allows us to view the fundamental frequency as well as the duration of each frequency. As expected, the fundamental frequency decreases, resulting in a lower pitch, as the liquid increases. Figures 2.10 and 2.11 show that the empty wineglass has a fundamental frequency of 1.89 kHz for a duration of about 1.15 seconds while the half full glass has 1.58 kHz for approximately 1.5 seconds.

The fundamental frequency duration increases as the volume rises; however, at about the volume halfway point, it begins to decrease. This requires additional analysis in future work but we believe that this may be due to the convergence constraint in the added mass operator that the added mass be below some maximum percentage of total mass.

**Performance:** the wineglass 3D model contained about 17,000 vertices, taking less than 30 seconds of precomputation to run our sound synthesis algorithm for fluid-structure coupling on a 2.40 GHz Lenovo Intel Core i7-4700MQ machine. Since our method enhances an existing pipeline by pre-processing steps

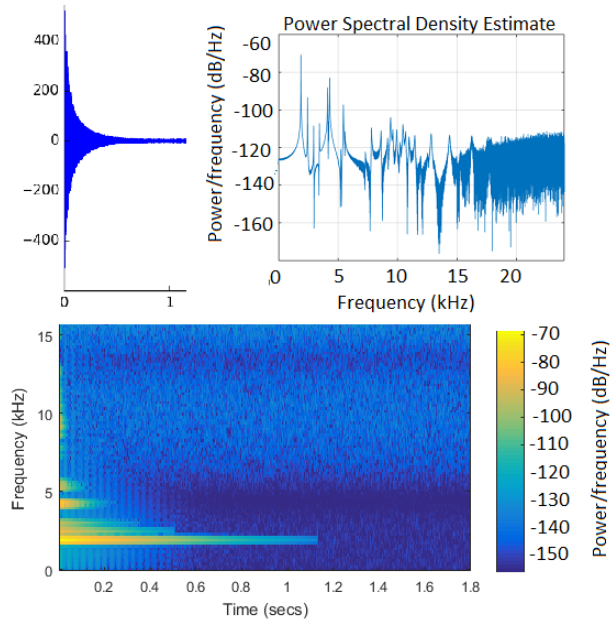


Figure 2.10: Empty wineglass. Top left: time domain signal. Top right: power spectral density estimation graph. Bottom: spectrogram for empty wineglass; higher frequency, higher pitch, shorter duration

only, we refer to related work for a computational cost analysis on the interactivity of sound synthesis techniques (Raghuvanshi and Lin, 2006).

Table 2.3: Solid and liquid are represented as tetrahedral meshes to detect boundary nodes and modify their mass elements although particle-based methods may also be used

Descriptor	Fluid	Structure
Object	Inviscid	Impermeable
Mesh	Tetrahedral	Tetrahedral
No. Vertices	~2-5k	~17k
No. Boundary Nodes	~500-1.5k	~500-1.5k

## 2.6 Applications

We integrated our prototype implementations with two applications: simulated liquid xylophone and interaction with virtual environments.

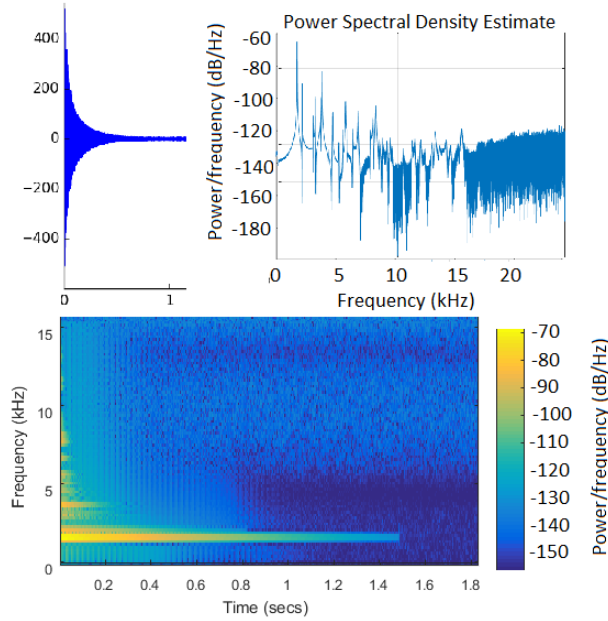


Figure 2.11: Half full wineglass. Top left: time domain signal. Top right: power spectral density estimation graph. Bottom: spectrogram for half full wineglass; lower frequency, lower pitch, longer duration

### 2.6.1 Simulated Liquid Xylophone

Modal analysis is performed before the simulation so the modes are pre-computed, allowing a user to interactively play the simulated liquid xylophone and generate sounds based on object material, liquid type, and liquid volume. Future additions could include making the simulated xylophone even more interactive by allowing the user to modify the type of liquids and/or the amount at run-time.

### 2.6.2 Integration With Virtual Environments

To demonstrate the plethora of non-empty objects that we interact with, we applied our method to a virtual kitchen. Next to each empty object is a filled object for sound comparison. Please see the supplementary video to hear the added sound effects.

## 2.7 Conclusion and Future Work

We have presented a novel method that extends the rigid-body sound synthesis pipeline to account for the change in sound frequency and duration resulting from various types and amounts of liquids that an object contains. Although our experiments show results in the general direction of how the frequency



Figure 2.12: User strikes each glass interactively in a virtual environment using a mouse click or virtual reality controller to play an octave of a simulated liquid xylophone



Figure 2.13: Users can interactively hear the sounds of various objects due to varying amounts and types of liquids in a virtul kitchen scene

Table 2.4: Generalizations for different solids and liquid volumes where Syn Freq is the synthesized frequency, in kHz

Solid	Liquid	Vol	$\rho(kg/m^3)$	Syn Freq
Porcelain Bowl	Water	Empty	1,000	2.9419
Porcelain Bowl	Water	1/2	1,000	2.7810
Metal Teapot	Water	Empty	1,000	10.0070
Metal Teapot	Water	1/2	1,000	7.5150
Wineglass	Water	Empty	1,000	1.8911
Wineglass	Water	1/2	1,000	1.5828

changes and produce similar sounds, our work assumes that liquids are inviscid, remain steady, and are not mixed. Our method should be extensible to handle mixed fluids. Liquids are also approximated as spheres to calculate added mass. The granularity of the solid mesh discretization also influences the results since the modifications to the mass matrix occur at the level of the mesh nodes. Other future research directions may include investigation of acoustic transfer and harmonic pressure (James et al., 2006a; Zheng and James, 2011a), as well as user evaluation on auditory perception of these added sound effects.

In summary, our sound synthesis algorithm for fluid-structure coupling adds pre-processing steps to the rigid body sound pipeline to simulate sound based on the volume and type of liquid contained within an object. The pre-computed steps are to identify and modify the mass matrix elements of the structural mesh nodes, along the domain of the fluid-structure boundary with an added mass operator proportional to the liquid’s density and volume. Since our technique adds mass from the liquid to the structural object before the simulation, the fluid(s) can be occluded as often is the case in graphical rendering and may be used in interactive 3D graphics and VR systems. Finally, we have demonstrated the effectiveness of our method for simulated musical instruments and composition, as well as enhanced realism of “glass-half-full” sounds in virtual environments.

## CHAPTER 3: ANALYZING LIQUID POURING SEQUENCES VIA AUDIO-VISUAL<sup>1</sup>

This chapter presents novel audio-based and audio-augmented, multimodal convolutional neural networks (CNNs), to estimate poured weight, perform overflow detection, and classify liquid and target container. Our Pouring Sequence Neural Networks (PSNNs) use the sound from a pouring sequence—a liquid being poured into a target container to improve classification accuracy for different environments, containers, and contents of the robot pouring task. They are trained and tested using the Rethink Robotics Baxter Research Robot. To the best of our knowledge, this is the first use of audio-visual neural networks to analyze liquid pouring sequences by classifying their weight, liquid, and receiving container.

### 3.1 Introduction

For robots to perform tasks individually or collaboratively, their ability to sense objects and substances in their environment is critical, especially when pouring liquids. Robots are increasingly performing more complicated human tasks, such as household activities, warehouse placements (e.g. Amazon Picking Challenge (Correll et al., 2018)), and other detection, recognition, and motion-planning tasks. Many methods for performing these robotic tasks use, and often primarily rely on, visual feedback and human interaction.

In this work, we propose using auditory cues to enhance learned feedback for robots in liquid pouring tasks. Audio has been used in robotics for localization of the spatial position of a sound source (Rascón and Ruiz, 2017), navigation (Huang et al., 1999), autonomous systems (Martinson and Schultz, 2009), sensorimotor learning (Boyer, 2015), and locomotion control (Ozkul et al., 2012), to name a few. Here, we investigate using sound to enhance a robot’s ability to estimate poured weights and types of liquids and containers. Humans are able to roughly sense a change in pitch when filling up a container (Lawson, 1965), and we demonstrate that robots can learn to do the same. With audio-visual neural networks, we classify weight, pouring contents, and containers for robot pouring tasks.

---

<sup>1</sup> This chapter previously appeared as an article in IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). The original citation is as follows: Justin Wilson, Auston Sterling, and Ming Lin. Analyzing liquid pouring sequences via audio-visual neural networks. pages 7702–7709, 11 2019b. doi: 10.1109/IROS40897.2019.8968118



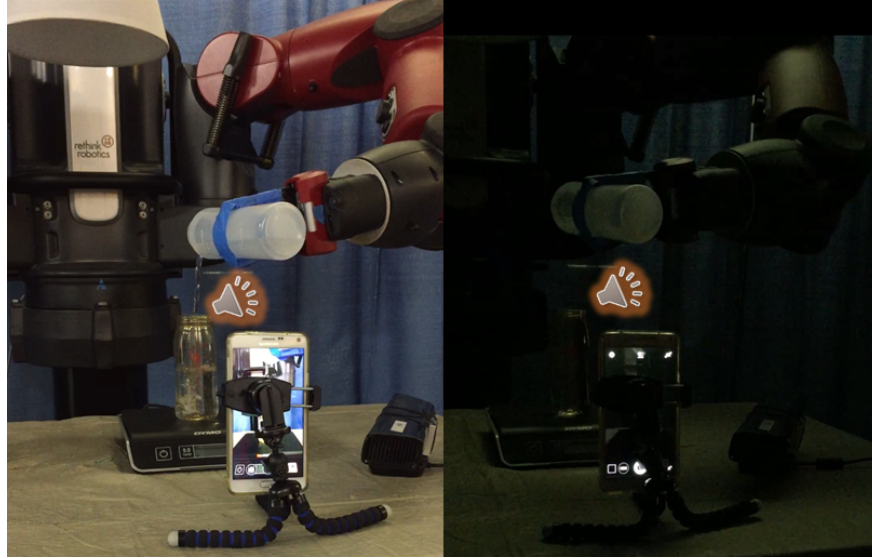


Figure 3.1: Our audio-augmented approach performs weight estimation, overflow detection, and content and container classification in bright environments (left) whereas our audio only based approach can be used in dark and occluded environments (right). Pouring sequences are recorded using either a smartphone or Microsoft Kinect’s built-in microphone array. Training data is generated by assigning digital scale measurements to discrete audio intervals and tested in experiments using Baxter robot and human experimenter pouring sequences. Various contents (water, rice, soda, and milk) and target containers (glass measuring cup, metal cup, Polyphenylsulfone (PPSU) bottle, plastic bottle, plastic cup, and square bowl) were evaluated.

Until recently, pouring tasks have often used predefined source amounts of a liquid. Now, (Clarke et al., 2018) demonstrates flow and weight estimation from audio-frequency mechanical vibrations of a robot scooping up and pouring granular materials and (Schenck and Fox, 2017) controls pouring with closed-loop visual feedback. Our motivation is to use audio to augment a robot’s visual sensing, thereby enabling the use of learned audio-visual feedback. To the best of our knowledge, this is the first use of learned audio-visual feedback to estimate the weight of poured liquids and classify liquid type and container.

The key contribution of this work is a novel, multimodal CNN for weight estimation, overflow detection, and liquid and container classification. We analyze liquid pouring sequences using audio and audio-visual variants of our neural network. We demonstrate their ability to compensate for vision-based challenges such as occlusion and transparency by evaluating on pairs of liquids and containers with hold out pouring sequences for both robot and human experimenter pouring. Our contributions are summarized as follows:

1. Training, validation, and test data generated from audio recordings and video images with ground truth measurements from a digital scale;
2. Audio-based convolutional neural network for multi-class weight estimation and binary classification for overflow detection by robotic systems;
3. Audio-augmented neural network enhancing the audio only based method with fused visual inputs for robots pouring contents into various target containers;
4. Pouring content and target container classification for robots, based on pouring sequence audio data.

### 3.2 Related Work

In this section, we discuss some of the state-of-the-art audio and video based classification techniques, focusing on temporal classification methods, motion planning, and learned estimation methods for the robot pouring task.

**Temporal classification methods:** these methods model the dependency, causality, and sequential nature of time series data such as audio. A number of temporal models have been introduced to represent this history and predict the likelihood of consecutive actions. Typical techniques include Hidden Markov Models (HMMs) (Rabiner, 1989), Conditional Random Fields (CRFs) (Lafferty et al., 2001), Recurrent Neural Networks (RNNs) (Jain and Medsker, 1999), and Long Short-Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997) networks.

Convolutional filters have also been used for temporal consistency; for example, WaveNet’s (van den Oord et al., 2016b) dilated causal convolutions and Temporary Convolutional Networks’ (TCNs) (Lea et al., 2017) dilated and encoder-decoder implementations. These models have in common the notion of convolution filters across time, computational speedup by updating time steps simultaneously rather than sequentially like recurrent networks, and frame-based classifications as a function of receptive fields (i.e. fixed-length periods of time). TCNs replace fully-connected layers with causal convolutional layers and sequential processing with parallel processing given the same filter in each layer. These characteristics along with state-of-the-art accuracy make TCNs a top choice for audio and visual classification tasks (Bai et al., 2018).



**Motion planning and monitoring:** while our work assumes specific robot and container placements, motion planning for pouring liquids focuses on motion going from start to end targets (Pan et al., 2016). To monitor pouring motion, sensory inputs from a chest-mounted camera and a wrist-mounted IMU sensor have been used (Wu et al., 2018). Related work has also categorized objects based on size, material, and other features (Griffith et al., 2012). For example, whether a container is fillable can be determined by using state sequences and a hierarchical spectral clustering algorithm (von Luxburg, 2007). This work is also relevant to our research by combining two modalities-sound and proprioception-to improve categorization accuracy.

**Learning based methods for pouring liquids:** (Clarke et al., 2018) is an audio based method that estimates the weight of granular material scooped. The technique is also used for pouring a desired material amount. The approach uses a recurrent neural network with convolutional layers and audio spectrogram input. A benefit of our multimodal approach is that the audio augments the visual data and sample intervals of the pouring sequence are evaluated independently (Table 3.2 for baseline comparisons). Analyzing the marginal benefit of recurrent layers in our neural networks is future work. Other learning based methods are based on human demonstrations (Yamaguchi et al., 2014, 2015). These methods model a variety of pouring motions involving shaking and using both robot arms.

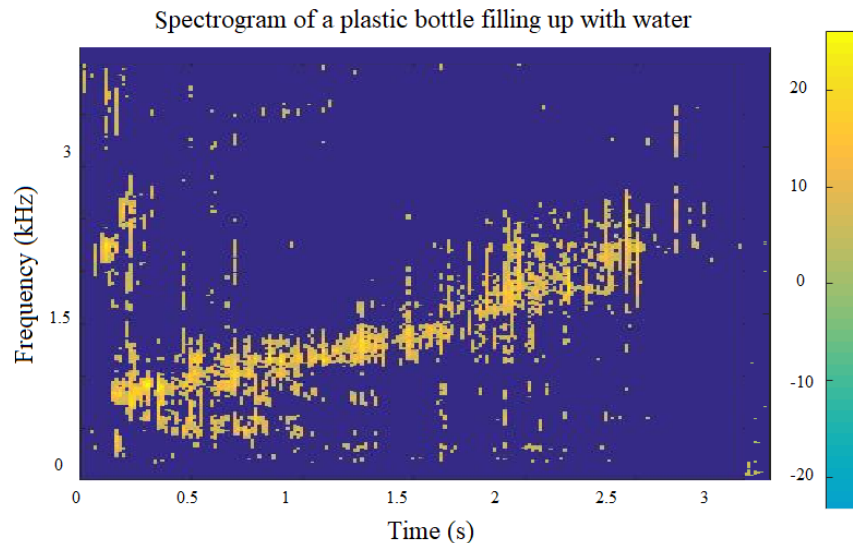


Figure 3.2: Spectrogram from a recorded pouring sequence. The frequency of a container filling up can be modeled based on its Helmholtz resonance (also referred to as a resonant cavity) (Webster and Davies, 2010). This resonant frequency increases over time as an object fills up with water as its cavity volume  $V_c$  decreases, supporting the use of an audio-based feature for the robot pouring task.

**Visual control for pouring liquids:** estimate liquid levels by identifying which pixels contain a liquid. (Schenck and Fox, 2017) uses a convolutional network to identify liquid pixels from RGB images and a second stage recurrent CNN-LSTM to estimate liquid volume. (Do et al., 2016) is a probabilistic approach using RGB-D to detect liquid levels. These estimation methods allow for the source container to carry amounts greater than that which the target container can receive because they can be used to control pouring without the need for specialized sensors.

### 3.3 Technical Approach

Our neural networks use audio and image data for weight estimation, overflow detection, and poured content and container classification, enhancing learning with sound alone or in conjunction with visual data. By augmenting visual data with sound, we can enhance a robot’s ability to detect and perform tasks with transparent or highly reflective containers and liquids in challenging and cluttered environments. To the best of our knowledge, this is the first use of an audio-augmented neural network to analyze liquid pouring sequences in robotics by estimating the weight of a pouring task and classifying poured contents and containers.

Our method allows for a source container to contain amounts greater than the capacity of the target container, as our Pouring Sequence Neural Networks (PSNNs) perform multiclass liquid, container, and weight classification and binary classification for overflow detection. Our audio-based approach uses a microphone for input, which can be found in any modern smartphone or Microsoft Kinect. Intervals of recorded audio are assigned a discrete weight class based on digital scale measurements for ground truth labeling. Training is performed offline, while classifications and overflow detection are the results of our neural network predictions.

#### 3.3.1 Task Overview

Our task is to use a mel-scaled<sup>2</sup> spectrogram of sound and video images of the target container to predict weight, liquid, and container at a point in time during a pouring sequence. A spectrogram is a two-dimensional representation of acoustic energy over frequency and time. Once target weight is reached or overflow detected, the robot can be signaled to stop pouring and return to its initial position. This task is

---

<sup>2</sup> The mel scale is a perceptually linear scale of pitch.

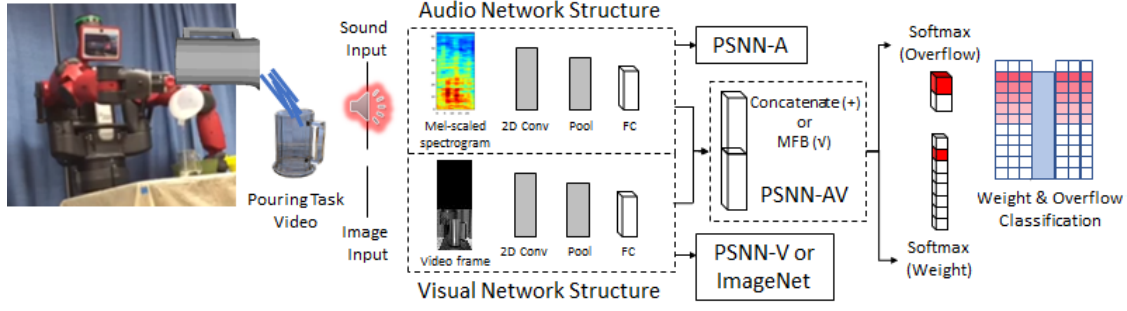


Figure 3.3: As the Baxter robot pours liquid from source to target container, a microphone records the audio of the target object filling up with liquid and a camera captures video images. The audio is split into 0.2 second intervals to match the digital scale sampling rate. These audio intervals are converted into mel-scaled spectrograms and passed through a multimodal CNN Pouring Sequence Neural Network (we refer to as PSNN) comprised of 2D convolutional, max pooling, fully connected, and softmax layers. Multi-class classification is used for discrete weight estimation (classes of 0.2 oz increments) and liquid and container prediction while binary classification is used for overflow detection. The network’s output may be used as a very simple stop command for the robot pouring task. Our method is trained on specific target container and content pairs.

more difficult than previous work in that it pours a specific amount rather than simply pouring the entire contents of the source. Moreover, our networks utilize audio information to augment a robot’s visual data. The use of audio features are reinforced by the change in audible frequency during a pouring sequence, known as the Helmholtz resonance.

### 3.3.2 Audio Feature Analysis: Helmholtz Resonance Frequency

As depicted in Fig. 3.2, the audio frequency increases as a container fills up with liquid, forming the basis of an audio-based feature for weight estimation and overflow detection. This increase in frequency can be modeled based on the Helmholtz resonance (also referred to as a resonant cavity) (Webster and Davies, 2010). This resonant frequency,  $f_{res}$  is calculated as:

$$f_{res} = \frac{c}{2\pi} \sqrt{\frac{s_p}{V_c l_p}}, \quad (3.1)$$

where  $f_{res}$  is proportional to the speed of sound in a gas  $c$  and square root of the cross section area  $s_p$  of the container neck, divided by cavity volume  $V_c$  and neck length  $l_p$ . When an object or liquid of volume  $V_p$  is placed/poured into the container, the cavity volume  $V_c$  decreases by that amount. By substituting  $V_c - V_p$  for  $V_c$ , then we can solve for poured volume  $V_p$  given  $V_c$ ,  $f_{res}$ , and corrected port  $l'_p$  (Rayleigh, 1945).

Audio	Example pouring sequence			
	Weight Est		Overflow	
	Truth	Pred	Truth	Pred
0.2s	0.0	0.0	NotFull	NotFull
0.4s	0.0	0.0	NotFull	NotFull
0.6s	0.0	0.0	NotFull	NotFull
0.8s	0.0	0.0	NotFull	NotFull
1.0s	0.1	0.0	NotFull	NotFull
1.2s	0.4	0.2	NotFull	NotFull
1.4s	1.0	0.8	NotFull	NotFull
1.6s	1.5	1.6	NotFull	NotFull
1.8s	2.7	2.4	NotFull	NotFull
2.0s	4.2	4.2	NotFull	NotFull
2.2s	5.8	6.4	NotFull	NotFull
2.4s	7.0	7.2	NotFull	Full
2.6s	9.0	8.6	NotFull	Full
2.8s	11.0	10.6	Full	Full
3.0s	11.8	11.4	Full	Full
3.2s	11.8	11.8	Full	Full

Table 3.1: Ground truth and predicted labels for a pouring sequence with intervals of 0.2, 0.5, and 1 second; 0.2 sec performed best. As length increases, there’s a larger variation of weight and frequencies for each training example.

$$V_p = V_c - \frac{s_p}{l'_p \left( \frac{2\pi f_{res}}{c} \right)^2} \quad (3.2)$$

While the resonant frequency adds justification for an audio-based network feature, it assumes the container itself will be symmetric, uniform width, and of a similar shape. Thus, we implement neural network based classifications that are trained on specific container and liquid pairs with holdout pouring sequences to relax some of these constraints.

### 3.3.3 Dataset Generation

We recorded 500 pouring sequences in total, for six target containers of varying material and geometry, each with three liquids and rice. Each container-liquid combination consisted of 20 pouring sequences. 3 hours of audio and video was captured to use 22,239 samples of 0.2 sec. Data was captured using an iPhone, Android, and Microsoft Kinect. Both robot and human experimenter pouring was performed.

For poured weight estimation, digital scale measurements were captured at a rate of 5 readings per second and synchronized to the audio and video recordings. The audio sampling rate was 256 kb/s and the

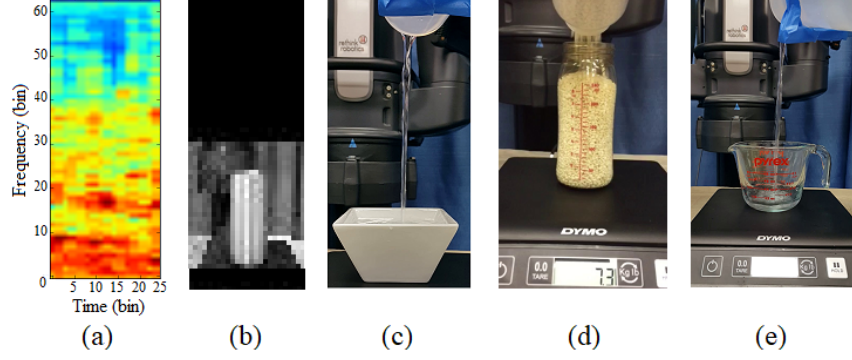


Figure 3.4: Audio-visual inputs 2D mel-scaled spectrogram (a) and cropped grayscale image (b). For opaque objects (c), visual information may be occluded. In these cases, PSNN-A outperforms PSNN-V and PSNN-AV. For transparent containers (d-e), our PSNN-V and PSNN-AV networks are able to detect visual deviations for both opaque (d) and transparent (e) pouring contents. The robot arm and digital scale LED are cropped out of images as to not influence network learning (b).

video frame rate was 30 per second. Digital scale readings were visible in the video and used for ground truth verification. However, since these video images were also an input into our audio-augmented network, they were cropped to remove the digital scale display and robot arm as to not influence training. For overflow detection, pouring sequences used for training were stopped at the time of overflow to assign full labels to the last few seconds of audio and the remaining intervals as not full. For both weight and overflow prediction, ground truth labels were assigned to discrete 0.2 sec intervals (or frames) for audio and visual data. Fig. 3.3 describes our neural network structure and Table 3.1 shows an example pouring sequence.

### 3.3.4 Neural Network Architecture of Audio-based Method

Our audio-based neural network model, also referred to as Pouring Sequence Neural Network (PSNN-A) shown in Fig. 3.3, is trained on mel-scaled spectrograms for audio intervals at the digital scale sampling rate of 0.2 seconds. A single convolutional layer followed by two dense layers with feature normalization performs optimally on our classification tasks (Table 3.2). We use consecutive full classification labels to indicate when to stop pouring for overflow detection. Section 3.4 covers our experiments and results against baseline methods. Section 3.5 offers analysis and insights into our audio-based (PSNN-A) and audio-augmented (PSNN-AV) convolutional neural networks.

**Audio input:** two audio input forms were considered – they are a 1D raw audio data and a 2D mel-scaled spectrogram. Using spectrograms as audio input has been shown to reduce over-fitting and improve accuracy (Huzaifah, 2017a). They are computed using a short-time Fourier transform with a Hann win-

dow of 2048 samples and an overlap of 25%. Frequency and time axes are downsampled and mapped into 64 mel-scaled frequency bins and 25 time bins to match the logarithmic perception of frequency (Sterling et al., 2018). We downsample the mel-spectrogram audio input and use a convolution kernel with an increased frequency resolution to reduce over-fitting.

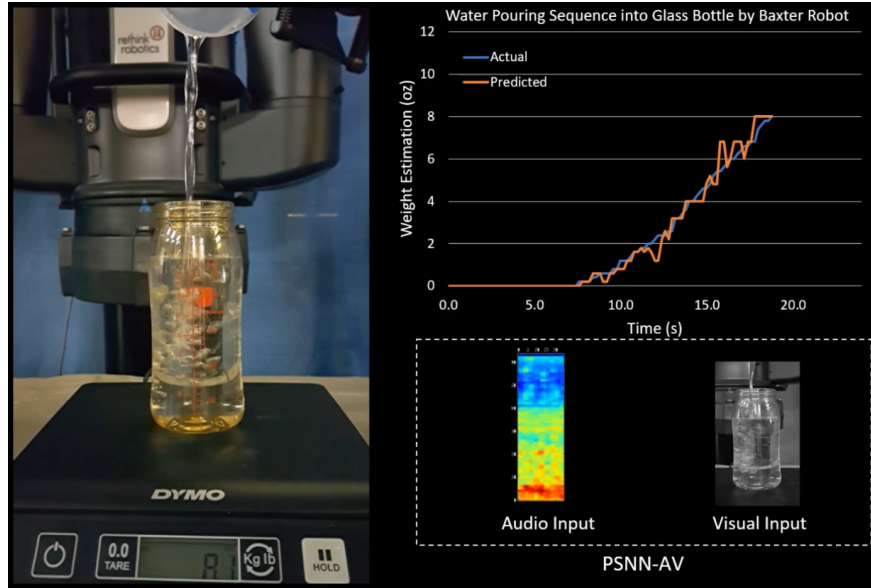


Figure 3.5: Demo video of liquid weights predicted by our PSNN neural network for a robot pouring sequence. (Left) video. (Top Right): actual versus predicted weights over time. (Bottom Right): audio and visual neural network inputs. Supplemental materials available at <http://gamma.cs.unc.edu/PSNN/>

### 3.3.5 Neural Network Architecture of Audio-Visual Method

The input size for audio and visual data have equivalent sizes (25 by 64 pixels). The inputs were designed this way to highlight the importance of estimating weight by changing vertical dimensions of frequency for audio and height for images respectively. Visualizations of inputs that maximize activation illustrate these distinguishing features (Fig. 3.9). Equivalence by concatenating inputs or fusing based on a bilinear model (Yu et al., 2017b) also allows the network to appropriately weight audio, visual, and audio-visual, given transparent or opaque target containers and contents.

**Visual input:** for our visual and audio-augmented networks, video images from a mobile device were assigned to corresponding audio intervals and digital scale recordings. To improve training and the effectiveness of our classification, visual data was augmented using techniques discussed in (Perez and Wang,

Weight Estimation by Method for Robot and Human Experimenter Water Pouring Sequences

Method	Input	PPSU Bottle, Robot Pour, N=20			PPSU Bottle, Human Pour, N=20		
		+/- 0.4 oz	Ave Err	Overflow	+/- 0.4 oz	Ave Err	Overflow
kNN (Cover and Hart, 1967)	A	66.4%	1.9 oz	71.9%	54.2%	2.7 oz	62.5%
Linear SVM (Bottou, 2010)	A	4.6%	3.8 oz	50.0%	13.6%	4.3 oz	50.0%
SoundNet5 (Aytar et al., 2016)	A	46.0%	1.9 oz	50.0%	42.4%	3.6 oz	50.0%
SoundNet8 (Aytar et al., 2016)	A	11.2%	3.3 oz	50.0%	29.2%	4.7 oz	50.0%
TCN (Lea et al., 2017)	A	78.4%	0.9 oz	50.0%	40.1%	3.7 oz	50.0%
<b>PSNN-A (Ours)</b>	A	<b>88.0%</b>	<b>0.5 oz</b>	<b>78.1%</b>	<b>75.8%</b>	<b>1.9 oz</b>	<b>64.3%</b>
ImageNet (Deng et al., 2009)	V	<b>83.8%</b>	<b>0.3 oz</b>	—*	<b>71.2%</b>	<b>0.4 oz</b>	—*
<b>PSNN-V (Ours)</b>	V	79.9%	0.6 oz	—*	66.5%	0.6 oz	—*
<b>PSNN-AV Cat (Ours)</b>	AV	<b>91.5%</b>	<b>0.2 oz</b>	—*	<b>86.4%</b>	<b>0.2 oz</b>	—*
<b>PSNN-AV MFB (Ours)</b>	AV	88.8%	<b>0.2 oz</b>	—*	71.2%	2.1 oz	—*

Table 3.2: Multiple models (**ours is PSNN**) and baselines were evaluated for audio and audio-visual based liquid pouring analysis. PSNN-AV correctly classified weight within 0.4 oz for up to 91.5% for robot and 86.4% of the human pouring sequences, outperforming all audio- and visual-only methods. This resulted in an average error of 0.2 oz and 0.2 oz respectively. \* Only audio-based neural networks were evaluated for overflow as visual information oversimplified the task.

2017) such as cropping. Correctly aligning the multimodal inputs with different sampling rates was also important as to not degrade neural network performance.

### 3.3.6 Implementation Details

All models were implemented with Tensorflow (Abadi et al., 2016) and Keras (Chollet, 2015). Parameters were learned using categorical cross entropy loss with Stochastic Gradient Descent. Training was performed using ADAM (Kingma and Ba, 2015) and run with a batch size of 64, with remaining hyperparameters tuned manually based on a separate validation set before final test set evaluation. Only audio-based methods were evaluated for overflow detection as incorporating visual information oversimplifies the task. Since there are fewer Full examples in a pouring sequence, audio data was balanced by randomly selecting an equal number of Full/Not Full audio intervals. Our datasets are available to aid future research in this area.

## 3.4 Results

We compared our method against baselines by conducting quantitative experiments on a variety of target containers, liquids, and rice. All baselines are trained on the same input data in order to provide a fair

Combined Robot and Human Experimenter Pouring Sequences				
Method	Input	Combined Container Dataset, N=40		
		+/- 0.4 oz	Ave Err	Overflow
kNN (Cover and Hart, 1967)	A	58.8%	2.4 oz	77.1%
Linear SVM (Bottou, 2010)	A	12.7%	4.0 oz	60.4%
SoundNet5 (Aytar et al., 2016)	A	21.2%	3.3 oz	50.0%
SoundNet8 (Aytar et al., 2016)	A	35.4%	4.4 oz	50.0%
TCN (Lea et al., 2017)	A	49.6%	2.6 oz	50.0%
<b>PSNN-A (Ours)</b>	A	<b>80.8%</b>	<b>1.3 oz</b>	<b>83.3%</b>
ImageNet (Deng et al., 2009)	V	68.1%	1.1 oz	—*
<b>PSNN-V (Ours)</b>	V	<b>78.0%</b>	<b>0.4 oz</b>	—*
<b>PSNN-AV Cat (Ours)</b>	AV	82.0%	0.3 oz	—*
<b>PSNN-AV MFB (Ours)</b>	AV	<b>86.7%</b>	<b>0.2 oz</b>	—*

Table 3.3: Evaluation results for the combined container dataset.

comparison. Deep network SoundNet (Aytar et al., 2016) is included as a commonly known sound-based classifier but it requires much more data to train. Pouring sequences were randomly divided into 80% training and 20% test sets. All target containers and pouring contents were included in training. Test data was based on hold out pouring sequences, which were removed from training and used only for testing.

### 3.4.1 Data Capture and Training

Video was recorded using a Samsung Galaxy Note 4 running Android 6.0.1, iPhone 6, and Microsoft Xbox 360 Kinect Sensor. Training was performed using a TITAN X GPU running on Ubuntu 16.04.5 LTS.

### 3.4.2 Pouring Sequence Experiments

Our experiments contained both human experimenter and robot pouring sequences. While robot pouring was varied by adjusting source container volume, experimenter pouring sequences offered additional variability, e.g. unfixed starting positions. All of our robot experiments were performed on a Rethink Robotics Baxter Research Robot, shown in Fig. 3.1. Pouring consisted of experimenters using both hands to hold the source container for human pouring and the Baxter robot’s 7 DOF left arm for robot pouring sequences. We used the Dymo Digital USB Postal Scale for ground truth weight estimates and a Samsung



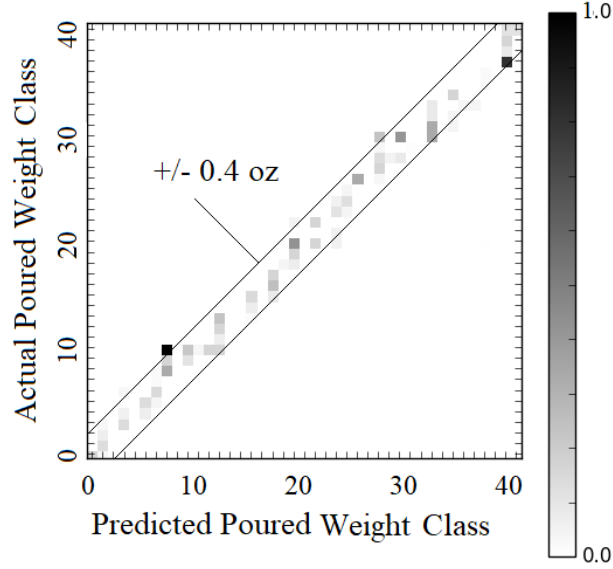


Figure 3.6: **PSNN-AV**: confusion matrix comparing actual to predicted poured amounts by classes of 0.2 oz (about 6 ml) weight increments. Class 0 represents empty; Class 1, 0.2 oz; and so on. Using audio and visual improves accuracy, especially at the beginning and end of the pouring sequence. Our system achieves up to **91.5%** (Table 3.2) and **91.2%** (Table 4.2) classification accuracy to within  $\pm 0.4$  oz using PSNN-AV.

Galaxy Note 4 for video recording.<sup>3</sup> According to the digital scale’s user guide, its accuracy is  $\pm 0.2$  oz when under and  $\pm 0.4$  oz when over half its capacity respectively.

For robot experiments, the target container rests on a tabletop, positioned slightly to the side and below the source container. The source container is fixed to the robot gripper and is pre-filled with an amount not known to the robot but greater than the amount required to fill the target container.

After a pouring sequence is initiated, audio from the target container filling up is recorded with a smartphone. Each audio interval is transformed into a mel-scaled spectrogram and input into our neural network model for weight and overflow classification. Once the desired pour amount is classified or overflow is detected, the robot can be signaled to stop the pouring sequence and return to its initial position.

### 3.4.3 Our PSNN Accuracy vs. Baseline Results

As illustrated in Table 3.2 and Fig. 5.8, up to 91.5% of the audio intervals for the robot pouring sequence into a PPSU bottle were classified to a weight class within 0.4 oz using our audio-augmented convolutional neural network (PSNN-AV); likewise, 86.4% of the human pouring sequence. This resulted

<sup>3</sup> Audio and video was also captured using an iPhone 6 and Microsoft Xbox 360 Kinect Sensor with built-in microphone array for comparison.

in an average error of 0.2 oz and 0.2 oz respectively. We also performed an evaluation on a combined pouring dataset containing both robot and human pouring sequences to explore the opportunity for transfer learning. A detailed analysis of these results will be discussed in Section 3.5.

Table 3.4, Table 4.2, and Fig. 3.7 demonstrate our method’s ability to be trained on different liquids and types of containers, including asymmetric objects. First, our audio-based PSNN-A network outperforms all baseline methods for audio only input. Second, when pouring content is visible, audio-augmented (PSNN-AV) outperforms audio-based (PSNN-A). This is especially true for more viscous liquids, such as milk, which make less noise during a pouring sequence.

Pl. Bottle	+/- 0.2 oz	+/- 0.4 oz	+/- 0.6 oz
Milk	57.8%	63.9%	68.4%
Rice	49.1%	64.4%	73.0%
Soda	73.0%	82.9%	88.4%
Water	69.6%	77.2%	84.0%

Table 3.4: Various pouring contents were evaluated. Rice was most difficult to precisely predict within +/- 0.2 oz.

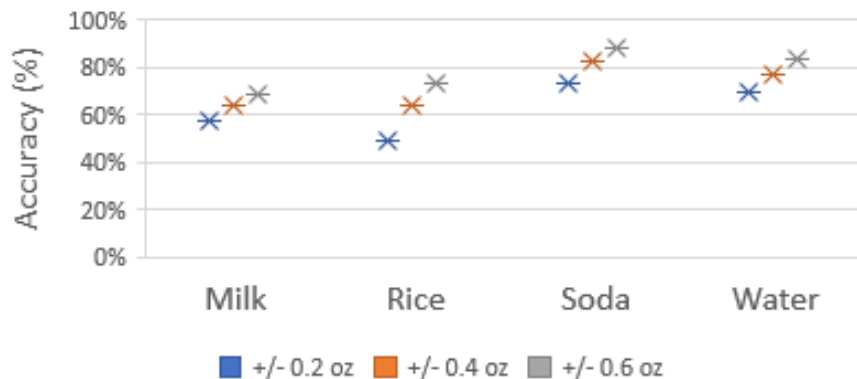


Figure 3.7: Various pouring contents evaluated with PSNN-AV. This graph displays the percentage of classified weights within +/- 0.2 oz (blue), 0.4 oz (orange), and 0.6 oz (gray) of ground truth. For instance, soda and water weights were easier to estimate than rice and milk.

We should note, however, that due to the relatively small size of the training set, our neural networks work well for target container and pouring content pairs that are described in this paper. Since all liquid-container pairs are included in training with hold-out pouring sequences, future work is needed for generalization to unseen and untrained target containers or pouring contents.

Accuracy by Method, Input, and Target Container for Robot Pouring Sequences				
Method	In	Transparent Plastic Cup Water +/-0.4 oz/Err	Transparent Glass Meas. Cup Water +/-0.4 oz/Err	Opaque Porcelain Bowl Water +/-0.4 oz/Err
kNN	A	34.7% / 3.4 oz	25.9% / 3.6 oz	48.1% / 2.2 oz
Linear SVM	A	5.4% / 3.4 oz	8.0% / 4.8 oz	8.9% / 3.3 oz
SoundNet5	A	14.0% / 3.4 oz	5.3% / 4.2 oz	6.4% / 4.4 oz
SoundNet8	A	11.6% / 3.2 oz	20.5% / 6.1 oz	9.4% / 3.5 oz
TCN	A	50.0% / 1.5 oz	39.5% / 1.9 oz	43.0% / 2.0 oz
<b>PSNN-A (Ours)</b>	<b>A</b>	<b>59.1% / 1.2 oz</b>	<b>46.8% / 1.2 oz</b>	<b>60.9% / 1.3 oz</b>
ImageNet	V	64.5% / 0.6 oz	51.7% / 1.2 oz	29.4% / 3.9 oz
<b>PSNN-V (Ours)</b>	<b>V</b>	<b>79.8% / 0.3 oz</b>	<b>63.9% / 0.5 oz</b>	<b>36.2% / 2.7 oz</b>
<b>PSNN-AV Cat (Ours)</b>	<b>AV</b>	<b>79.0% / 0.3 oz</b>	<b>70.0% / 0.4 oz</b>	40.0% / 3.4 oz
<b>PSNN-AV MFB (Ours)</b>	<b>AV</b>	69.2% / 0.4 oz	44.9% / 1.7 oz	<b>42.6% / 2.6 oz</b>

Table 3.5: Multiple network models and baselines were evaluated. **Ours is PSNN**. Headings indicate distinguishing properties being evaluated. The PSNN networks outperform baseline networks on the same type of inputs, while the multimodal PSNN-AV network outperformed each independent modality.

Classification Accuracy and Average Error Continued				
Method	In	Opaque Metal Cup Water +/-0.4 oz/Err	Transparent PPSU Bottle Milk +/-0.4 oz/Err	Transparent PPSU Bottle Rice +/-0.4 oz/Err
kNN	A	41.0% / 2.5 oz	38.2% / 2.7 oz	48.4% / 1.7 oz
Linear SVM	A	7.0% / 4.1 oz	33.2% / 3.5 oz	12.8% / 2.3 oz
SoundNet5	A	4.4% / 4.7 oz	9.7% / 3.0 oz	9.6% / 2.4 oz
SoundNet8	A	13.1% / 4.2 oz	13.4% / 5.8 oz	8.8% / 3.4 oz
TCN	A	51.5% / 1.7 oz	34.0% / 3.9 oz	52.7% / 1.7 oz
<b>PSNN-A (Ours)</b>	<b>A</b>	<b>65.9% / 0.7 oz</b>	<b>45.0% / 1.8 oz</b>	<b>74.1% / 1.0 oz</b>
ImageNet	V	20.0% / 6.1 oz	65.1% / 0.4 oz	77.0% / <b>0.4 oz</b>
<b>PSNN-V (Ours)</b>	<b>V</b>	<b>25.3% / 4.6 oz</b>	<b>68.9% / 0.4 oz</b>	<b>83.7% / 0.4 oz</b>
<b>PSNN-AV Cat (Ours)</b>	<b>AV</b>	48.5% / 1.9 oz	71.8% / 0.4 oz	<b>91.2% / 0.2 oz</b>
<b>PSNN-AV MFB (Ours)</b>	<b>AV</b>	<b>65.5% / 1.2 oz</b>	<b>82.4% / 0.2 oz</b>	81.8% / 0.3 oz

Table 3.6: Varying type of liquid poured, multiple network models and baselines were evaluated for weight estimation of robot pouring sequences.

Classification Accuracy for Pouring Content via Human  
Pouring and Target Container via Robot Pouring

Pl. Bottle	Content %	Water	Container %
Milk	86.5%	Plastic Bottle (0)	99.6%
Rice	79.6%	Metal Cup (1)	88.4%
Soda	72.4%	PPSU Bottle (2)	69.2%
Water	97.9%	Glass Measuring Cup (3)	64.2%
		Porcelain Square Bowl (4)	61.3%
		Plastic Cup (5)	78.5%

Table 3.7: PSNN-A predicts pouring content and target container with high accuracy, learning features from audio to correctly classify liquid and container from pouring sequence data.

### 3.4.4 Liquid and container classification

Table 3.7 highlights PSNN-A’s ability to classify liquid and target container from pouring sequence audio. Higher accuracy can be achieved by excluding intervals before and after pouring when audio is not present, or by using PSNN-AV. For future work, we plan to investigate if accuracy varies over time. For instance, is content classification accuracy higher in the beginning of a pouring sequence?

We concluded our testing with an ablative analysis for hyper-parameter optimization (e.g. training epochs, interval length, etc.). Our pouring sequence dataset with audio and visual data is made available to support future research and evaluation in this area of robotics.

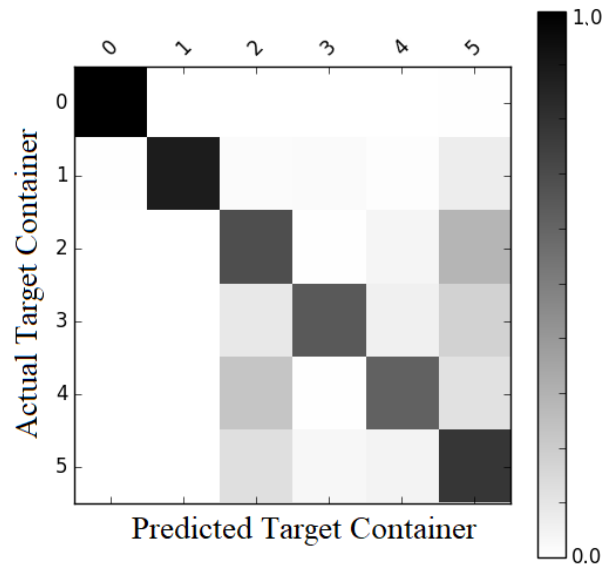


Figure 3.8: Confusion matrix of actual and predicted container classifications based on audio-only pouring sequences. It shows PSNN-A learning to classify between objects of the same material (e.g. Plastic Bottle and Cup) and same type (Plastic Bottle and PPSU Bottle). 0-5 labels in Table 3.7.

### 3.5 Analysis

In this work, we implement multimodal neural networks based on audio and visual data to the robotic task of weight estimation for pouring a liquid, overflow detection, and liquid and container classification. Our PSNN neural networks outperform existing methods in the experiments that we have performed. Our contributions include new audio-visual datasets and multimodal neural network architectures designed for the robot pouring task. In this section, we analyze the improved performance of using our methods.

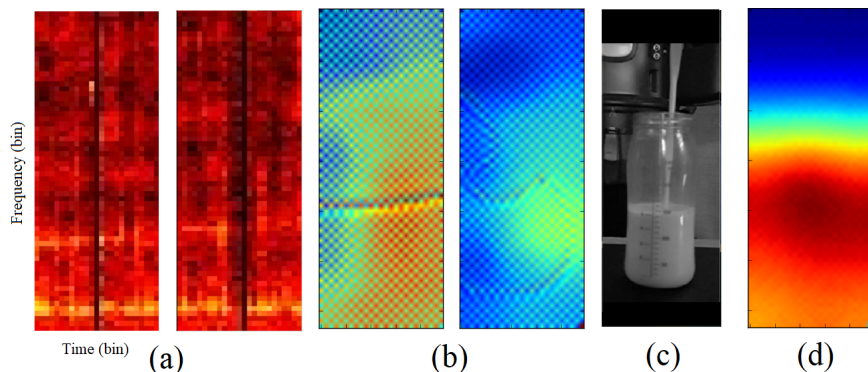


Figure 3.9: **Audio activations:** example pouring sequence spectrograms (a). Audio inputs that would maximize our audio-based neural network activation for a couple of specific weights (b). This demonstrates PSNN-A’s ability to learn changes in frequency to distinguish between weight classes. **Visual activations:** example grayscale, cropped visual input (c). Visual input that would maximize the activation of our visual neural network (d). This shows PSNN-V’s ability to learn visual features for distinguishing between classes for visible pouring contents (Fig. 3.4).

#### 3.5.1 Activation Maximization Visualizations

We analyzed activation maximizations to visualize the spectrogram audio and visual input which would produce the highest activation for a given volume class. Fig. 3.9 shows activation maximization for the audio-based PSNN-A network as additional volume is poured (a-b) and the visual-based PSNN-V network (c-d). Both highlight the importance of audio (frequency) and visual (height) respectively.

#### 3.5.2 Model Comparisons

For opaque target containers, the audio only PSNN-A performs the best compared to PSNN-V and PSNN-AV due to occlusion. For transparent target containers, multimodal PSNN-AV provides the maximum classification accuracy and minimum average error. Even for a quiet, viscous liquid like milk, aug-

menting visual data with audio outperformed audio or visual only with 82.4% accuracy and 0.2 oz average error compared to 45.0% and 68.9% respectively. (Table 4.2).

#### **3.5.2.1 PSNN-A Normalized**

Normalizing the features allows for a more symmetric optimization between frequency and time given a mel-scaled spectrogram input. Scaling is important to normalize the differences in feature scale. When feature scaling is not applied, then gradient descent may require a smaller learning rate to ensure that the optimization converges and does not over step the minimum.

#### **3.5.2.2 PSNN-A and Temporal Convolutional Networks (TCN)**

Our methods outperform time distributed baselines because while the pouring task is sequential, it does not rely as heavily on previous inputs since each 0.2 second spectrogram encodes the current state. Furthermore, time distributed methods may overfit and fail to cover more general and inconsistent pouring behavior. PSNN can evaluate inputs independently since each mel-scaled spectrogram already encodes historical information given a frame-based interval.

#### **3.5.2.3 Robot and Human Poured**

Given an equal number of training examples and epochs, robot pouring sequences are more accurate than human poured (Table 3.2). In other words, robot pouring sequences require less data and training time because of more uniform pouring sequences, producing more consistent audio and visual data for each weight class. Additional analysis of the impact pouring rates have on accuracy will be further investigated in future work.

#### **3.5.2.4 Combined Pour Dataset**

For TCN and PSNN, the combined dataset of robot and human pouring sequences mostly performs medially as compared to each separately (Table 3.2). For PSNN-V, however, additional visual data of a combined dataset performs better with 0.4 oz average error compared to 0.6 oz for both robot and human pouring. This implies visual data is less affected by pouring consistency than audio, benefiting from additional yet mixed data.

### 3.5.2.5 Interval Length

Audio sampling intervals of 0.2, 0.5, and 1 second were evaluated. 0.2 is the minimum based on the digital scale sampling rate. Faster intervals performed better, which is to be expected since the interval is assigned a single ground truth weight and smaller time intervals would represent a smaller change in poured amount over that time. As the length increases, the interval likely has a larger variation of frequencies for each training example.

## 3.6 Conclusion and Future Work

We present novel, audio-based and audio-augmented neural networks to estimate poured weight, perform overflow detection, and classify pouring liquid and target container based on pouring sequence audio-visual data. By recording the sound of the pouring sequence as the target container fills up, an audio-based feature can be applied to different containers and liquids for the robot pouring task. Our method is trained on specific target container and content pairs using both human and robot pouring sequences and is tested on the Baxter robot. We also evaluate our dataset on a combined container dataset and make our audio-visual data available for future research. To our knowledge, this is the first use of audio-visual neural networks to analyze liquid pouring sequences by classifying weight, liquid, and target container.

**Future Directions:** to increase accuracy beyond current performance, we plan to analyze augmentations of our audio data with environmental, room acoustics, and other alterations. As the task involves temporal data, sequential layers can be introduced into the neural network model. This may be especially helpful for audio only PSNN-A classification at the beginning and end of pouring sequences when there are no pouring sounds. In addition, we can compare against lower-dimensional parameterizations of the sound such as audio features like spectral centroid, skew, kurtosis, and rolloff. Comparison with model-based methods when target container geometry is known may shed new insight as well.

Our current neural networks do not generalize to unseen target containers or pouring contents. We plan to research ways to generalize our approach, which may involve multitask learning, increasing the size of our training set, adding more audio and visual data augmentations, or incorporating synthetic pouring sequences. Using a multiple output neural network rather than separately trained neural networks for poured weight, content, and target container classification may also help as well as using a ratio of volume over the target container volume or a combination of all of the above.

Finally, we will explore if our approach can be applied to other granular materials and liquids in addition to rice and the liquids that we've tested to date. Furthermore, we plan to evaluate if container size and function (e.g. fillable or not) can be determined by using the spectral hierarchical clustering algorithm (von Luxburg, 2007) or PSNN to categorize objects based on size, material, and other features (Griffith et al., 2012).



## CHAPTER 4: AUDIO-VISUAL OBJECT TRACKING FOR MULTIPLE OBJECTS<sup>1</sup>

This chapter describes an audio-visual object tracking (AVOT) neural network that reduces tracking error and drift by using audio of the impact sounds from object collisions, rolling, etc. It may be used in conjunction with other neural networks to augment visually based object detection and tracking methods. Using the synthetic Sound-20K audio-visual dataset, AVOT outperforms single-modality deep learning methods, when there is audio from object collisions.

### 4.1 Introduction

Deep learning has enabled state-of-the-art techniques for image classification and object detection in images and video (Liu et al., 2016; Redmon et al., 2015a; Ren et al., 2015c). Object tracking classifies bounding boxes for each object in a video over time. These methods are useful for applications in autonomous driving (Geiger et al., 2012), mobile robotics (Schulz et al., 2001), person tracking (Checka et al., 2001), speaker recognition (Spors et al., 2001; Qian et al., 2019), and 3D reconstruction (Prisacariu et al., 2015). For more granularity beyond bounding boxes, object segmentation provides pixel-level annotations (Voigtlaender et al., 2019; Perazzi et al., 2016). These existing object trackers achieve real-time performance and continue to improve on accuracy and the number of classes that they can detect.

However, occlusion, similar object categories, and smaller object sizes remain a challenge for visually based trackers (Liu et al., 2016). Auditory cues can assist in these exacting areas, especially when similar and/or smaller objects are of a different material (Aytar et al., 2016). In this paper, we propose an audio-visual object tracker (AVOT) that augments visual only trackers with fused audio in a jointly trained end-to-end model. It is evaluated using synthetic Sound-20K dataset (Zhang et al., 2017c), consisting of tabletop sized objects of different geometry and materials. The data contains videos with multiple objects of various shapes (e.g. bottle, knife, etc.) and materials (e.g. steel, wood, etc.) colliding in a virtual scene.

---

<sup>1</sup> This chapter previously appeared as an article in the International Conference on Robotics and Automation (ICRA). The original citation is as follows: Justin Wilson and Ming C. Lin. Avot: Audio-visual object tracking of multiple objects for robotics. 2020a

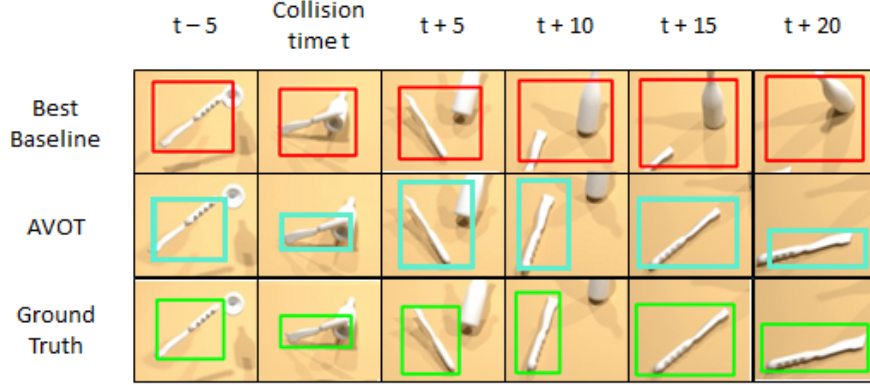


Figure 4.1: An example failure case improved by our audio-visual object tracker. (Top row) best baseline, CSRT in this case, incorrectly latches to the wrong object after collision. (Middle row) our AVOT method continues to correctly track the object post-collision. (Bottom row) ground truth annotated by the experimenter. For clarity, we show the bounding box for only one of the objects being tracked, although the methods track both objects. Please see the Supplementary Video for more demonstration.

Colliding includes objects colliding within the scene, with each other, rolling, etc. We use videos with one, two, or three colliding objects.

Other than speaker recognition, this is the first use of an audio-visual neural network for tracking tabletop sized objects and enhancing visual object trackers. The key contributions of this work include:

- An end-to-end, jointly trained audio-visual object tracker (AVOT) to enhance visual object tracking;
- Ground truth bounding box annotations for Sound-20K audio-visual dataset with 1, 2, and 3 object scenes;
- Scheduler for object detection re-initialization based on audio onset detection when using multi-modal tracking.

Fusing audio with visual data, AVOT achieves 77.7% IoU post-collision tracking accuracy compared to 68.6% IoU using deep-learning visual tracking, SSD– (Liu et al., 2016), and 38.4% using CSRT (Lukezic et al., 2017) for virtual scenes with multiple objects based on our annotated Sound-20K dataset of 19 tabletop sized object classes of varying geometry and materials.

## 4.2 Background and Related Work

While object detection methods must search over the entire search space to first detect an object, tracking algorithms can be much faster by leveraging knowledge from previous frames to reduce the search

Object Detection/Tracking Datasets		
Dataset	# Class	# Img/Vid
COCO (Lin et al., 2014b)	80	330K img
DAVIS (Caelles et al., 2019)	384	10.5K img
ILSVRC (Russakovsky et al., 2015)	1000	1.4M img
KITTI (Voigtlaender et al., 2019; Milan et al., 2016)	8	10.9K img
OTB (Wu et al., 2015a)	100	100 vid
PASCAL VOC (Everingham et al.)	20	21K img
Sound-20K (Zhang et al., 2017c)	55+	20K vid
VOT2018 (Kristan et al., 2016)	35	147K img
YouTube-VOS (Xu et al., 2018)	7800+	4K+ vid

Table 4.1: In contrast to other datasets, the Sound-20K dataset contains the largest audio-visual data for object interactions in a virtual scene and provides an excellent baseline for assessing the accuracy of our AVOT method against others.

space. However, this can make error recovery difficult. Tracking is also more complex because unlike object detection bounding boxes which are class specific, tracking is object specific. It assigns identifiable bounding boxes to each object and attempts to maintain each assignment over all frames. So, tracking not only detects but also maintains bounding box assignment for each object, over all frames. More granular than bounding boxes, segmentation may also be performed for pixel-level annotations. For an attribute and performance comparison, object tracking benchmark (Wu et al., 2015a) provides attribute and performance comparisons between various methods and evaluation criteria.

The majority of object detection and tracking methods are visually based, even though some datasets are generated from videos with audio. Table 4.1 lists commonly used datasets for object detection and tracking evaluation. We add ground truth bounding box annotations to the Sound-20K dataset and use it as a baseline for assessing the accuracy of our AVOT method. While the general methodology of research areas such as speaker detection and person tracking leverage both audio and visual information, their implementations are specifically aimed at tracking human speakers (e.g. face detection is part of their pipeline). Our method aims to be applicable in a broader context and does not make assumptions about the targets. It currently can track up to nineteen object-material classes. Next we discuss object detection, tracking, and audio-visual techniques in more detail, as compared to our work.

### 4.2.1 Object Detection

In addition to overall classification of an image, researchers are interested in also detecting and classifying the specific objects within an image. This can be achieved by using object detection methods to locate and label each object with a class-specific bounding box. As is similar in image classification, object detection techniques require large amounts of training data but in its case, more annotations for each example. Because, in the case of object detection, training data requires both class labels and bounding box coordinates for each object. For example, the PASCAL Visual Object Classes (VOC) dataset contains images, object annotations, and segmentations for twenty different classes. Other available datasets are mentioned in Table 4.1. Unfortunately, only a few datasets make available the video and accompanying audio, making audio-visual methods more time-consuming to explore. We contribute our Sound-20K ground truth annotations to aid future audio-visual research in this area.

**Video object detection:** object detection can be performed not only on images but on video as well. Here, additional contextual information is available such as sound and image sequence. This temporal memory has allowed video detection to achieve start-of-the-art performance and speeds by learning lightweight scene features for mobile (Liu et al., 2019) and shifting channels along the dimension of time (Lin et al., 2019). However, video also introduces new challenges such as motion blur, defocus, and various poses. Temporal coherence can also be used to overcome these defects with flow-guided feature aggregation (Zhu et al., 2017), for instance. Finally, in addition to scene features, time shifting, and temporally coherent features, temporal propagation for on demand detection has also yielded efficiency gains (Chen et al., 2018).

### 4.2.2 Object Tracking

Object tracking differs from object detection in that the labels and bounding boxes are dependent. In other words, tracking attempts to establish correspondences of the same object over multiple frames, for example, one particular car in traffic over time. While object tracking has been studied for decades, numerous factors remain a challenge, such as illumination variation, occlusion, and background clutters (Wu et al., 2015a). Given the sequential nature of the task and method, tracking can be fast and efficient but also accumulate error and drift. Moreover, it is not easy for object trackers to recover from failure or an incorrect assignment to another object. Approaches such as frame skipping, Siamese trackers, and deep

learning are a few of the existing techniques being used to perform object tracking and segmentation quickly and accurately.

**Frame skipping:** this baseline approach *detects* every N-th frame and *tracks* frames in between. Advantages of frame skipping are realized in terms of speed and precision by efficiently tracking on a majority of the frames while still allowing for correction by performing detection every N keyframes. This can be referred to as a fixed scheduler. Dynamic schedulers have also been considered. For instance, Detect or Track (Luo et al., 2019) uses a scheduler network to determine whether to detect or track at certain frames. In our research, we propose an audio-based scheduler procedure that can alter tracking during audio onset detection in videos.

**Siamese trackers:** Siamese neural networks take two inputs and, with shared weights, predict if the two inputs belong to the same output class. Fully-convolutional Siamese approaches can be used for object tracking (Bertinetto et al., 2016; Li et al., 2018; Zhu et al., 2018; He et al., 2018; Yang and Chan, 2018) and unlike batch processing, online Siamese methods can perform tracking on streaming video with access only to current and previous frames (Wang et al., 2019). To improve initialization of online adaption-based deep networks such as these, offline meta-learning has been applied (Park and Berg, 2018). Asymmetric Siamese networks have also been studied and learn a linear template to search test images by cross-correlation (Valmadre et al., 2017).

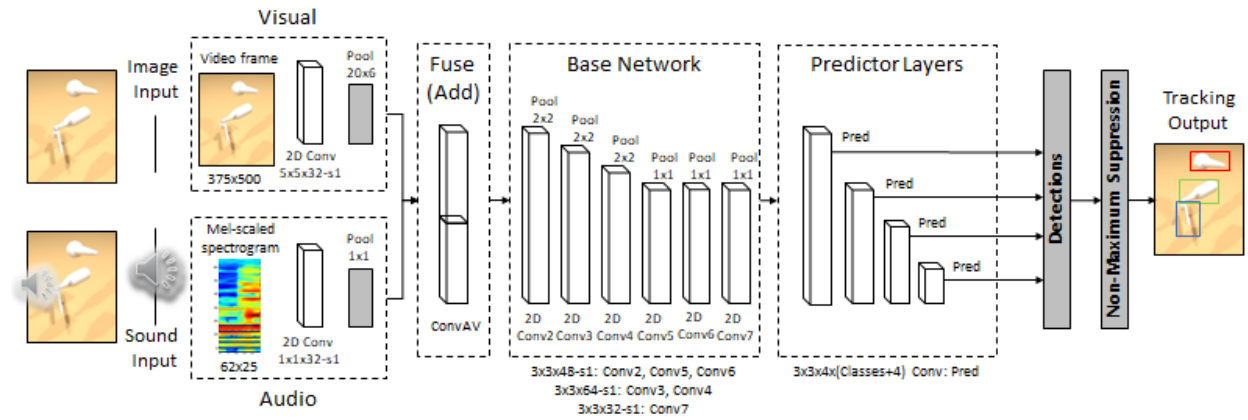


Figure 4.2: Audio-Visual Object Tracker (AVOT) neural network architecture. AVOT is a feed-forward convolutional neural network that classifies and scales a fixed number of anchor bounding boxes to track objects in a video. Here, we define an object based on its geometry and material. Convolutional layers from the visual and audio inputs are fused using an add merge layer before being input into a base network of convolutional layers similar to standard classification networks. The base is then followed by predictor layers for detection, as is done in SSD, however designed and optimized for our audio-visual dataset and task. The single best detection for each object is then selected using non-maximum suppression.

**Deep learning:** last but not least, object tracking performed using deep learning. Faster R-CNN (Ren et al., 2015a) is a real-time, state-of-the-art object tracker and four staged end-to-end neural network. First, a convolutional feature map of the image is obtained by extracting from a convolutional layer of a pre-trained CNN (e.g. ImageNet (Krizhevsky et al., 2012a), ResNet (He et al., 2016), MobileNets (Howard et al., 2017), DenseNet (Huang et al., 2017), etc.). The second stage is a Region Proposal Network, which are reference bounding boxes uniformly placed across the image. In this stage, specific regions are identified and adjusted based on the convolutional feature map from the first step. The third stage applies Region of Interest (RoI) Pooling to extract features from the convolutional map for each region. The fourth and final step then uses those features to classify the content in the bounding box (e.g. bottle, table, etc., background) and adjust the classified bounding box to a better fit, predicting  $\Delta x_{center}$ ,  $\Delta y_{center}$ ,  $\Delta width$ ,  $\Delta height$  from an anchor.

Single Shot MultiBox Detector (SSD) (Liu et al., 2016) is another real-time, state-of-the-art object tracker. SSD is slightly better than YOLO (Redmon et al., 2015a, 2016) in terms of speed while improving upon accuracy with additional feature layers on top of a base network<sup>2</sup>. Furthermore, SSD is slightly better than Faster R-CNN in terms of accuracy while eliminating object proposals with multiple feature maps of differing resolution. Although SSD uses similar default boxes, it applies them to several feature maps of different resolutions. In addition to a single unified framework for training and prediction, SSD input images are smaller at 300 x 300, compared to 512 x 512 for Faster R-CNN and 448 x 448 for YOLO. This enables faster processing over other single shot, region proposal, and pooling techniques. This permits a wider range of computer vision applications to leverage this architecture. We use SSD as both a baseline and base network for our AVOT tracker.

### 4.2.3 Audio-Visual Methods

Audio-visual techniques have been used for speech separation (Ephrat et al., 2018a), object and geometry classification (Zhang et al., 2017c,b; Sterling et al., 2018), and audio-visual correspondence learning (Arandjelovic and Zisserman, 2018, 2017). Most directly related to audio-visual object tracking is speaker recognition (Spors et al., 2001; Qian et al., 2019), tracking from audio-visual data using a linear prediction method (Anusha and Roy, 2015), and object detection and tracking with audio and optical sig-

---

<sup>2</sup> ImageNet VGG-16 was used as a base, but other neural networks should also produce good results.

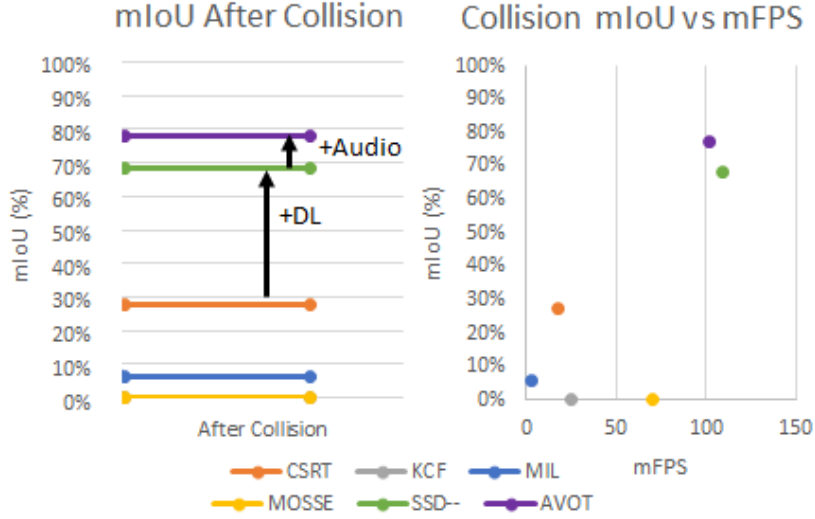


Figure 4.3: Existing object trackers performance decline when objects collide in a moving two object Sound-20K virtual scene whereas AVOT improves with audio onset. Post-collision (i.e. when there’s audio), deep learning (DL) methods achieve nearly 40% higher in accuracy over other methods and AVOT further outperforms SSD— by another 10% in *mean Intersection over Union* (mIoU) with an added benefit of audio-visual input. A scheduler network gated on audio can be used to achieve the best run performance and/or the highest accuracy across all cases using multimodal trackers.

nals (Holz). For speaker recognition, a face tracking algorithm and microphone array are used to estimate speaker position. These methods fuse audio and visual data by leveraging time delays in audio and motion changes in visual. While both modalities, in theory, can distinguish these changes, one may be more adept to do so. Also, the fusion of the two can decrease uncertainty and increase reliability (Ngiam et al., 2011b). Finally, audio can also come from contact microphones or acoustical sensors to capture touch sounds and optical signals for gesture recognition (Holz). In our approach, we leverage audio from impact sounds of objects and images from video.

### 4.3 Technical Approach

Unlike visually based object trackers, our method defines each object by its geometry and material. With audio-visual data, the same shape (e.g. bottle) with different materials (e.g. steel vs. wood) are distinguishable and are therefore considered to be different objects. Our work also considers colliding objects. While a challenge for visually based tracking methods (Fig. 4.3), they provide auditory cues for an audio-visual object tracker. Scheduling between trackers can then be enabled based on audio availability.

Given the location of an object in the first frame of video, the object tracking task is to quickly and accurately estimate its position in all successive frames (Smeulders et al., 2014). More specifically, for each video frame in a sequence  $F = f_1, f_2, \dots, f_N$  where  $N$  is number of frames, obtain bounding boxes  $B = b_1, b_2, \dots, b_M$  where  $M$  is the number of objects.

#### 4.3.1 AVOT Neural Network Architecture

Similar to existing object tracking architectures, AVOT is a feed-forward convolutional neural network that classifies and scales a fixed number of anchor bounding boxes to fit each object in an image. We define an object based on its geometry and material. AVOT leverages audio and visual data for a more granular definition of an object to distinguish between objects with the same appearance but different materials.

**Audio input:** Audio frames from Sound-20K (Zhang et al., 2017c) videos match the image frame rate of 33 frames per second. As a result, each jpeg image has a corresponding 29 ms audio wav file. The audio is converted to mel-scaled spectrograms and serve as the audio input given their performance in CNNs for other tasks (Huzaifah, 2017a). They are computed using a short-time Fourier transform with a 512 sample Hann window and 12.5% overlap. A Hanning (Hann) window was selected for its suitability for a variety of signals, good frequency resolution, and reduced spectral leakage. Each spectrogram is individually normalized and downsampled to a size of 62 frequency bins by 25 time bins (Fig. 4.4). Binning provides for appropriate fusion with image dimensions and weight matching to the logarithmic perception of frequency (Sterling et al., 2018).

**Image input:** image dimensions are 500 x 375 pixels. Since SSD evaluated input sizes 300 x 300 and 512 x 512 (YOLO 448 x 448), our images are augmented but input dimensions unmodified as they fall within range of previous work. For data augmentation, we use common image transformations and sampling strategy similar to SSD and YOLO. Random cropping can be especially useful for creating zoomed in and out training examples to aid the classification of small objects in PASCAL VOC and Sound-20K. Each training image randomly samples from a data augmentation sequence to make the model more robust to object size and shape (Liu et al., 2016). We use a reduced layer variation of VGG16 (Simonyan and Zisserman, 2015) as the base network leading up to our detection prediction layers. Images were



extracted from video using ffmpeg with CRF scale set to 0 (lossless) and libx264 set to vcodec (Zhang et al., 2017c). Each image is fused with its corresponding audio via an add-merge layer.

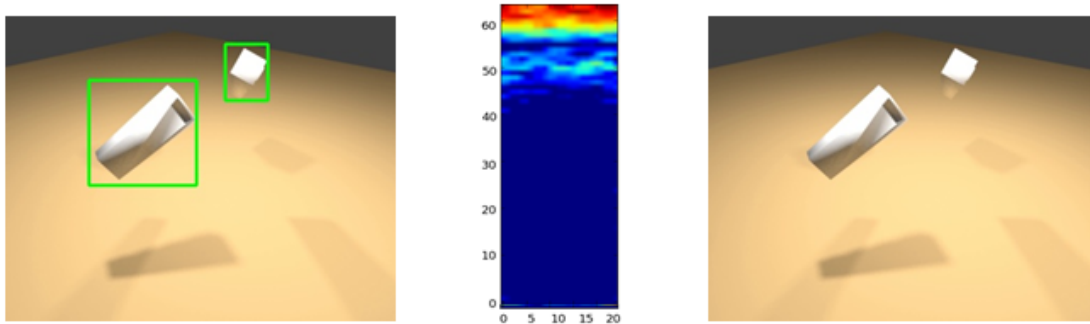


Figure 4.4: AVOT needs ground truth boxes (left), input audio from the scene video converted to a mel-scaled spectrogram (center), and input image (right) for each object during training. We predict shape offsets and confidences for all object categories where an object is defined by its geometry and material.

**Architecture:** Fig. 4.2 illustrates the layers of our multimodal object tracker neural network. The early visual layer is based on (LeCun and Bengio, 1998) and audio layer based on impact (Sterling et al., 2018) and environmental sound (Huzaifah, 2017a) classification. Convolutional layers from the visual and audio inputs are fused using an add merge layer. A multiply-merge layer was also considered and resulted in a similar training loss, however, at 1.5x the number of training epochs. Fused features are then input into a base network. Given our relatively small annotated audio-visual dataset, our base network is a reduced version of the standard image classification architecture (Krizhevsky et al., 2012a). The base is then followed by predictor, or also referred to as feature or classifier layers. Upper and lower feature maps are used for detection, as is done in SSD, to promote consistency and capture fine details respectively. The single best detection for each object is then selected using non-maximum suppression.

#### 4.3.2 AVOT Dataset

Ground truth annotations were manually labeled by the experimenter for 18 objects. Each object is unique by geometry and material. The dataset is comprised of 17 three second videos of 103 image and audio frames each. This resulted in a total of 1,752 audio and visual segments. Videos contained one, two, and three colliding objects per scene. Our training and test datasets are split 80% and 20% respectively. The test dataset randomly samples frames from each video that are held out from training and used only for evaluation. For example, a video with 100 frames will have 20 frames randomly selected for test and

the remaining 80 frames used for training. Fig. 4.5 shows loss by epoch for our AVOT tracker compared to a variation of visually based SSD.

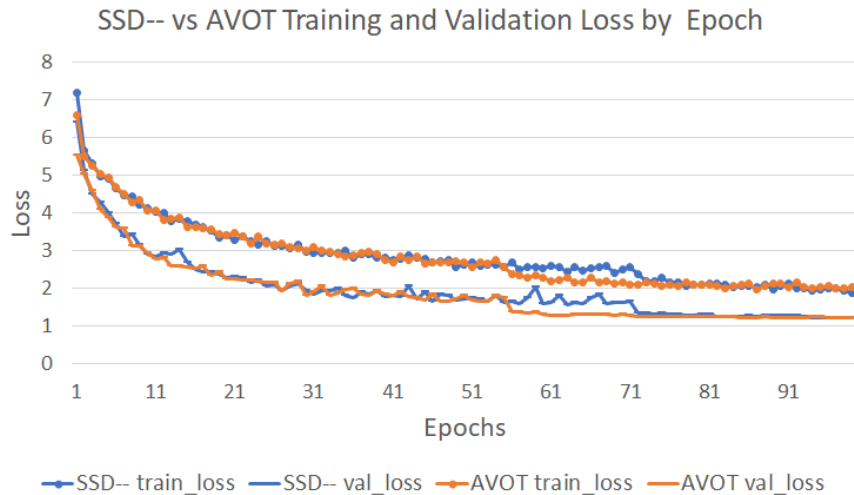


Figure 4.5: The training (circle) and validation (line) loss for SSD– (blue) and AVOT (orange). Multi-modal AVOT loss seems to decrease more consistently than visual only SSD with reduced layers, denoted as SSD–.

### 4.3.3 Implementation Details

All models were implemented with Tensorflow (Abadi et al., 2016) and Keras (Chollet, 2015). AVOT was run with early stopping at a maximum of 100 epochs, 100 steps per epoch, and batch size of 16 (Fig. 4.5). Training was performed using an Adam optimizer (Kingma and Ba, 2015) and loss as defined by the weighted sum of localization loss (Smooth L1) and confidence loss (Softmax). We use a reduced variation SSD for predictor layers. AVOT anchor box scaling factors were set to 0.08, 0.16, 0.32, 0.64, and 0.96 and aspect ratios 0.5, 1.0, and 2.0 (Liu et al., 2016). Here, we do not use SSD aspect ratios 1/3 or 3 given a smaller number of target classes. There are five scaling factors for four predictor layers because the last scaling factor is used for the second aspect ratio box of the last predictor layer. Although fewer layers, detections are still based on small 3 x 3 kernels at each feature map offset (Liu et al., 2016).

**Initialization:** our AVOT neural network uses `he_normal` initialization (He et al., 2015b). For evaluation, we also initialize baseline methods with `he_normal` rather than fine tune on pre-trained networks. Recent research suggests equivalent performance between random initialization for training instead of

pre-trained weights (He et al., 2019). Furthermore, given a smaller dataset of Sound-20K with ground truth annotations, we have reduced the layers of baseline implementations to avoid overfitting.

**Non-maximum suppression (NMS) (Neubeck and Gool, 2006):** object trackers may produce more than one overlapping bounding box that are greater than the confidence and IoU thresholds for the same object. NMS is a post-process that selects the bounding box with the greatest confidence and suppresses remaining bounding boxes that overlap this maximum by some threshold. Here, NMS confidence and IoU threshold are set to 0.5 (Liu et al., 2016).

**Scheduler network:** Impact sounds from objects colliding emulate a type of scheduler network that can improve detections post collision. For added efficiency, only visual inputs can be processed leading up to audio onset. After, both audio and visual inputs can be used. In the case of our synthetic dataset, there is no audio prior to collision which makes audio onset easier to detect than videos with noise.

#### 4.4 Experiments and Results

Evaluation was performed using ground truth annotations on the Sound-20K audio-visual dataset. This dataset is comprised of synthetic videos of multiple objects colliding in a scene. Training took roughly 30 minutes running on Ubuntu 16.04.6 LTS with a single Titan X GPU. We use Intersection over Union (IoU) for accuracy between ground truth and predicted object bounding boxes. As a general rule of thumb, a true positive prediction occurs when  $IoU \geq 0.5$ , according to the PASCAL VOC challenge. We measure the speed in mean frames per second (mFPS) with a batch size of 16 using a Titan X and cuDNN v7.4.2.

**OpenCV implementations:** online Multiple Instance Learning (MIL) (Babenko et al., 2009), Kernelized Correlation Filters (KCF) (Henriques et al., 2015), Discriminative Correlation Filter with Channel and Spatial Reliability (CSRT) (Lukezic et al., 2017), and an adaptive correlation filter known as Minimum Output Sum of Squared Error (MOSSE) (Bolme et al., 2010) are a few trackers available in OpenCV (Bradski, 2000). We selected to evaluate these as baselines due to their advantages in terms of accuracy and/or speed. For these methods, appearance is learned from first frame bounding boxes that are initialized with ground truth coordinates.

mIoU / mFPS Object Tracking Accuracy by Method		
Method	2 Objects	3 Objects
<b>AVOT (Ours)</b>	<b>58.3%</b> / 101.6	<b>66.1%</b> / 101.0
CSRT (Lukezic et al., 2017)	46.9% / 17.1	30.1% / 4.7
KCF (Henriques et al., 2015)	13.5% / 24.9	1.7% / 38.6
MIL (Babenko et al., 2009)	43.0% / 2.5	21.6% / 1.6
MOSSE (Bolme et al., 2010)	7.6% / 70.4	1.0% / <b>74.5</b>
SSD– (Liu et al., 2016)	55.5% / <b>108.7</b>	65.9% / <b>103.8</b>

Table 4.2: Multiple network models were evaluated on accuracy and time using mean Intersection over Union (mIoU) and mean frames per second (mFPS). **Ours is AVOT**. Failure cases for baseline methods without audio tend to classify to the correct geometry but wrong material. By exploiting both visual and audio data, AVOT achieves the highest level of tracking accuracy, with nearly comparable best runtime performance, over existing visual tracking methods.

#### 4.4.1 Our Results vs. Baselines

Given our limited number of training examples in our audio-visual dataset, we used a reduced layer implementation of SSD (labeled in Table 4.2 as SSD–) for a baseline and base network for AVOT. Our AVOT neural network outperforms SSD– and other baseline methods after collision (Fig. 4.3). As shown in Fig. 4.3, AVOT was able to achieve the highest level of accuracy of 80% in *mean Intersection over Union* (mIoU)— about 10% more accurate than SSD. While these results are AVOT only, we further propose a scheduler network (**Algorithm 1**) to switch between AVOT and other methods based on audio onset to maximize accuracy and performance over all frames in multimodal object tracking.

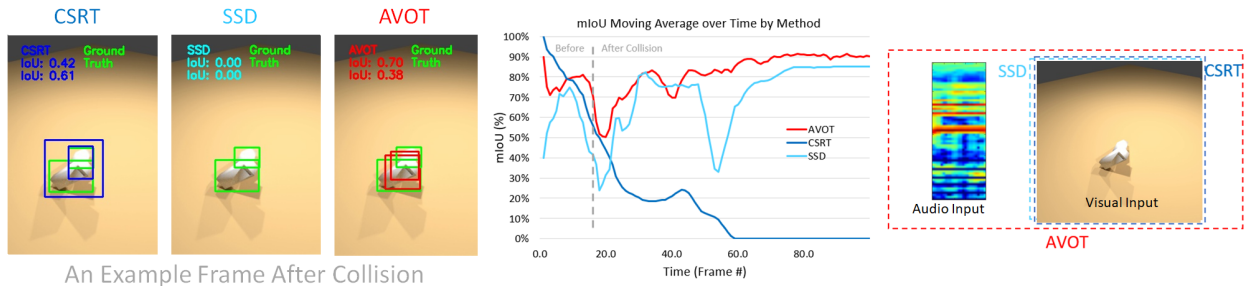


Figure 4.6: We compare CSRT and SSD to our AVOT method for multi-object tracking. Two colliding objects with the same geometry but different materials are tracked free-falling in a virtual scene from Sound-20K (Zhang et al., 2017c). CSRT is unable to maintain tracking post-collision and while SSD recovers, it temporarily loses tracking at the time of occlusion. Audio-visual AVOT maintains tracking across all frames. Please see the Supplementary Video for more demonstrations.

#### 4.4.2 Maximization Activation

We analyzed activation maximizations to visualize the spectrogram audio and visual input which would produce the highest activation for a given volume class. They demonstrate features being learned by both modalities for the object tracking task. Please see the Supplementary Video for demonstration.

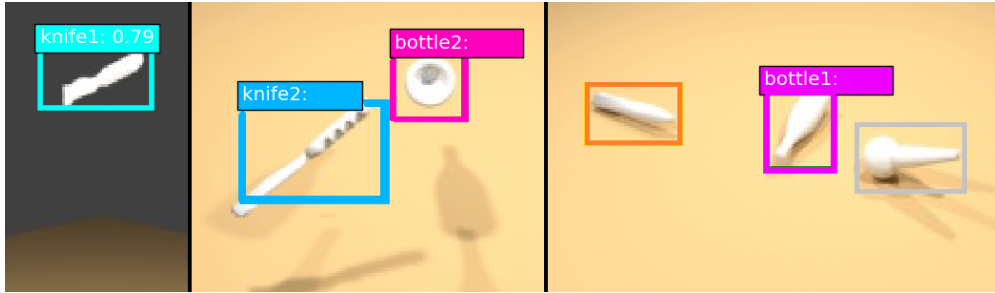


Figure 4.7: Examples of AVOT applied to virtual scene of Sound-20K with predicted bounding box. These are exemplary screenshots of AVOT performing object tracking before and after collisions for one, two, and three object virtual scenes. Notice alphanumeric labels (e.g. bottle1 and bottle1) to differentiate the same geometry with different materials.

#### 4.5 Conclusion

We present AVOT, an end-to-end trained neural network for object tracking using audio and visual data from videos. To distinguish between similar objects with different materials, we define an object based on its geometry and material. This more granular categorization benefits from a multimodal learning approach using audio and visual data, where audio is typically available from the sources of video but are currently underutilized. By fusing audio with visual data, our audio-visual object tracker (AVOT) outperforms single-modality methods when audio is present from impact, collision, and rolling sounds while maintaining real-time performance. We evaluated against Sound-20K and make our audio-visual data along with ground truth bounding box annotations available for future research in this area.

**Future work:** we will expand the size of our training set by annotating more objects in the Sound-20K dataset, increase the number of object classes that we are predicting, evaluate alternative fusion methods, and perform sensitivity analysis on scaling factors and aspect ratios. We would also like to augment our audio data and experiment with a variation of our object tracker with audio only.

## CHAPTER 5: AUDIO-AUGMENTED SCENE RECONSTRUCTION ON MOBILE DEVICES<sup>1</sup>

This chapter describes echoreconstruction, an audio-visual method that uses the reflections of sound to aid in geometry and audio reconstruction. The mobile phone prototype emits pulsed audio while recording video for RGB-based 3D reconstruction and audio-visual classification. Reflected sound and images from the video are input into our audio (EchoCNN-A) and audio-visual (EchoCNN-AV) convolutional neural networks for surface and sound source detection, depth estimation, and material classification. The inferences from these classifications enhance scene 3D reconstructions containing open spaces and reflective surfaces by depth filtering, inpainting, and placement of unmixed sound sources in the scene.

### 5.1 Introduction

Reconstruction techniques have enabled significant contributions in detection (Lea et al., 2016), segmentation (Golodetz\* et al., 2015; Arnab et al., 2015), and semantic understanding (Song et al., 2017). They have also been used to generate large-scale, labeled datasets of object (Wu et al., 2015b) and scene (Dai et al., 2017a) geometric models to further aid training and sensing in a 3D environment. However, scenes containing open and reflective surfaces, such as windows and mirrors, can present a unique set of

---

<sup>1</sup> This chapter is currently under review.



Figure 5.1: *Left*: ground truth image. *Before* (*Middle*) and *after* (*Right*) audio-augmented rendering of an indoor scene with open and closed reflective surfaces. The reconstruction is enhanced by EchoCNN inferences of surface detection, depth estimation, and material classification based on audio-visual reflecting sound and image inputs.

challenges. First, they are difficult to detect and reconstruct due to their transparency and high reflectivity. Distinguishing between glass (e.g. window) and an opening in the space is an important part of the audio-visual experience. Finally, illumination, background objects, and min/max depth ranges can be confounding factors. While advances have been made to account for these challenging surfaces (Sinha et al., 2012; Whelan et al., 2018; Chabra et al., 2019), our work augments these state-of-the-art visual methods by adding an audio context of surface detection, depth, and material estimation.

Previous work has used sound to better understand objects in scenes. For instance, impact sounds from interacting with objects in a scene to perform segmentation (Arnab et al., 2015) and neural networks to emulate the sensory interactions of human information processing (Zhang et al., 2017d). Audio has also been used to automatically compute material (Ren et al., 2013b), object (Zhang et al., 2017d), scene (Schissler et al., 2018), and acoustical (Tang et al., 2020) properties. Better still, using both audio and visual sensory inputs has been shown to be even more effective; for example, multi-modal learning for object classification (Sterling et al., 2018; Wilson et al., 2019a) and object tracking (Wilson and Lin, 2020b).

Fusing multiple modalities, such as vision and sound, provide a wider range of possibilities than either single modality alone. In this work, we demonstrate that augmenting vision-based techniques with audio, referred to as “EchoCNN,” can detect open and reflective surfaces, its depth, and material, thereby enhancing 3D object and scene reconstruction. We highlight some key results below:

- EchoCNN, a fused audio-visual CNN architecture for classifying open/closed surfaces, their depth, and material;
- EchoReconstruction, a staged audio-visual 3D reconstruction pipeline that uses mobile phones to enhance scene geometry containing windows, mirrors, and open surfaces with depth filtering and inpainting based on EchoCNN inferences;
- Semantic rendering of window and mirror in audio-augmented reconstructions based on point of view (e.g. environment outside of the window or reflected view of a TV);
- Real and synthetic audio-visual ground truth data for multiple scenes containing windows and mirrors in addition to reflection separation data (direct, early, or late reverberations).



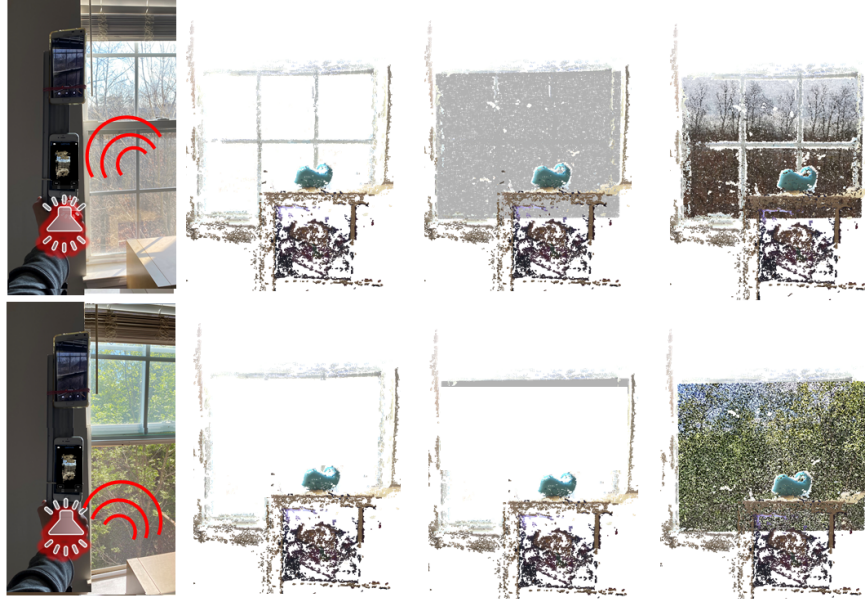


Figure 5.2: *Top row*: closed window in winter. *Bottom row*: opened in spring. *Column 1*: mobile echoreconstruction prototype; the bottom phone emits pulsed audio and performs a RGB-based 3D reconstruction (live (Tanskanen et al., 2013) or photogrammetric (Metashape, 2020)); the top phone records video. *Column 2*: initial reconstruction based on state-of-the-art commercially available Astrivis app. *Column 3*: our audio-visual EchoCNN convolutional neural network classifies open or closed surface, depth, and material. *Column 4*: semantic reconstruction of the window accounting for EchoCNN inferences.

## 5.2 Related Work

Previous research in 3D reconstruction, audio-based classifications, and echolocation are discussed in this section in addition to existing techniques for reconstructing open and reflective surfaces.

### 5.2.1 3D reconstruction

Object and scene reconstruction methods generate 3D scans using RGB and RGB-D data. For example, Structure from Motion (SfM) (Westoby et al., 2012), Multi-View Stereo (MVS) (Seitz et al., 2006), and Shape from Shading (Zhang et al., 1999) are all techniques to scan a scene and its objects. Static (Newcombe et al., 2011; Golodetz\* et al., 2015) and dynamic (Newcombe et al., 2015; Dai et al., 2017b) scenes can also be scanned in real-time using commodity sensors such as the Microsoft Kinect and GPU hardware. 3D scene reconstructions have also been performed with sound based on time of flight sensing (Crocco et al., 2016). Not only has this previous research generated large amounts of 3D scene (Silberman et al., 2012b; Song et al., 2017) and object (Singh et al., 2014; Lai et al., 2011; Wu et al., 2015b) data, they also benefit from these datasets by using them for training vision-based neural networks



Example 3D Reconstruction Methods	
Type	Methods
Active (RGB-D)	KinectFusion, DynamicFusion, BundleFusion
Passive (RGB)	SLAM, SFM, (Tanskanen et al., 2013), ScanNet, (Whelan et al., 2018)
Stereo	MVS, StereoDRNet
Lidar	(Kada and Mckinley, 2009)
Ultrasonic	(Zhang et al., 2017a)
Time of flight	(Crocco et al., 2016)

Table 5.1: 3D reconstruction methods by type such as passive (RGB), active (RGB-D), or other sensor (e.g. ultrasound, lidar, etc.); single or multiple views; and static or dynamic scenes.

for classification, segmentation, and other downstream tasks. Depth estimation algorithms (Eigen and Fergus, 2014; Alhashim and Wonka, 2018; Chabra et al., 2019) also create 3D reconstructions by fusing depth maps using ICP and volumetric fusion (Izadi et al., 2011).

#### 5.2.1.1 Glass and mirror reconstruction

Reflective surfaces produce identifiable audio and visual artifacts that can be used to help their detection. For example, researchers have developed algorithms to detect reflections in images taken through glass using correlations of 8-by-8 pixel blocks (Shih et al., 2015), image gradients (Kopf et al., 2013), and two layer renderings (Sinha et al., 2012). (Sutherland, 1968) used ultrasonic sensor logic to track continuous wave ultrasound and (Zhang et al., 2017a) to detect obstacles such as glass and mirrors by using frequencies outside of the human audible range. More recently, reflective surfaces have been detected by utilizing a mirrored variation of an AprilTag (Olson, 2011; Wang and Olson, 2016). (Whelan et al., 2018) use the reflective surface to their advantage by recognizing the AprilTag attached to their Kinect scanning device when it appears in the scene. Depth jumps and incomplete reconstructions have also been used (Lysenkov et al., 2012). However, vision based approaches require the right illumination, non-blurred imagery, and limited clutter behind the surface that may limit the reflection. We show that sound creates a distinct audio signal, providing reconstruction methods complementary data about the presence of windows and mirrors without additional sensors.

#### 5.2.2 Acoustic imaging and audio-based classifiers

We begin with an introduction into sound propagation, room acoustics, and audio-visual classifiers.

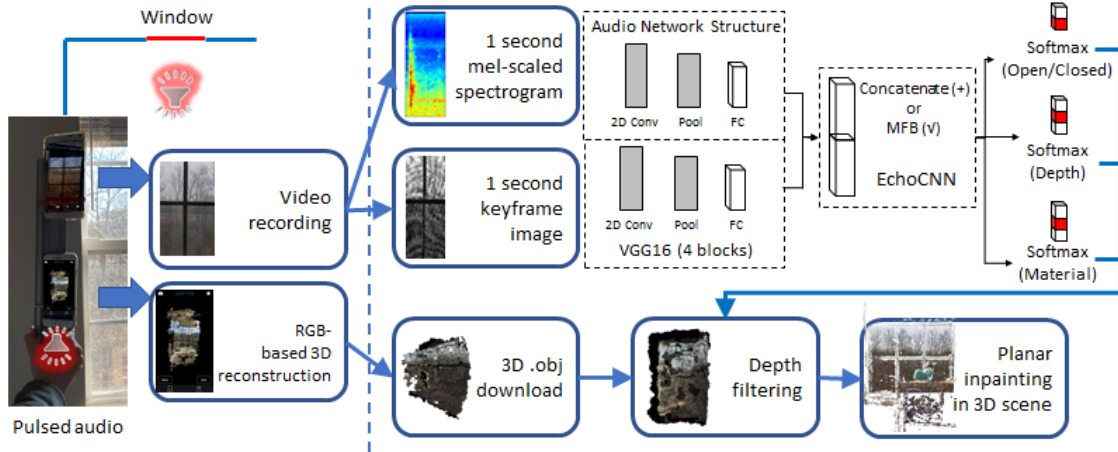


Figure 5.3: *Staged approach* to enhance scene and object reconstruction using audio-visual data. Our echoreconstruction prototype consists of two smartphones - one recording (top) and one emitting/reconstructing (bottom). As the bottom smartphone moves to reconstruct the scene and emits 100 ms pulsed audio (Section 5.3.3), the top smartphone is used to record video of the direct and reflecting sound. The receiving audio is split into 1.0 second intervals to allow for reverberation. These audio intervals are converted into mel-scaled spectrograms and passed through a multimodal echoreconstruction convolutional neural network (we refer to as EchoCNN) comprised of 2D convolutional, max pooling, fully connected, and softmax layers. EchoCNN classifications inform depth filtering and hole filling steps to resolve planar discontinuities in scans caused by reflective surfaces, such as windows and mirrors. Binary classification is used to predict if a window is open or closed. Multi-class classification is used for depth and material estimation.

**Acoustics:** various models have been developed to simulate sound propagation in a 3D environment, such as wave-based (Mehra et al., 2015a), ray tracing based (Rungta et al., 2016b), sound source clustering (Tsingos et al., 2004b), multipole equivalent source methods (James et al., 2006b), and single point multipole expansion (Zheng and James, 2011b), representing outgoing pressure fields. (Godoy et al., 2018) uses acoustics and a smartphone for an app to detect car location and distance from walking pedestrians using temporal dynamics. (Bianco et al., 2019) further discusses theory and applications of machine learning in acoustics. Computational imaging approaches have also used acoustics for non-line-of-sight imaging (Lindell et al., 2019), 3D room geometry reconstruction from audio-visual sensors (Kim et al., 2017), and acoustic imaging on a mobile device (Mao et al., 2018). To reconstruct windows and mirrors, our work uses room acoustics given the surface materials of the room (Schissler et al., 2018) and distance from sound source. However, prior work and downstream processes often require a watertight reconstruction which can be difficult to generate in the presence of glass. Our approach addresses these issues using an integrated audio-visual CNN that can detect discontinuity, depth, and materials.

**Audio-based classification:** using principles from sound synthesis, propagation, and room acoustics, audio classifiers have been developed for environmental sound (Gemmeke et al., 2017; Piczak, 2015; Salamon et al., 2014), material (Arnab et al., 2015), and object shape (Zhang et al., 2017d) classification. Audio input can take the form of raw audio, spectral shape descriptors (Michael et al., 2005; Cowling and Sitte, 2003; Smith III, 2020), or frequency spectral coefficients that we also adopt in our method.

**Audio-visual learning:** similar to its applications in natural language processing (NLP) and visual questing & answering systems (Kim et al., 2016, 2020; Hannan et al., 2020), multi-modal learning using both audio-visual sensory inputs has also been used for classification tasks (Sterling et al., 2018; Wilson et al., 2019a), audio-visual zooming (Nair et al., 2019), and sound source separation (Ephrat et al., 2018b; Lee and Seung, 2000) which have also isolated waves for specific generation tasks. Although similar in spirit, our audio-visual method, “Echoreconstruction” differs from the existing methods by learning absorption and reflectance properties to detect a reflective surface, its depth, and material.

### 5.3 Technical Approach

In this work, we adopt “echolocation” as an analog for our echoreconstruction method. According to (Egan, 1988), echo is defined as *distinct* reflections of the original sound with a sufficient sound level to be clearly heard above the general reverberation. Although perceptible echo is abated because of precedence (known as the Haas effect) (Long, 2014), returning sound waves are received after reflecting off of a solid surface. We use these distinct, reflecting sounds to design a staged approach of audio and audio-visual convolutional neural networks. EchoCNN-A and EchoCNN-AV can be used to estimate depth based on reverberation times (Fig. 5.9), recognize material based on frequency and amplitude, and handle both static and dynamic scenes with moving objects based on Doppler shift. All of which enhance scene and object reconstruction by detecting planar discontinuities from open or closed surfaces and then estimating depth and material.

#### 5.3.1 Echolocation

Echolocation is the use of reflected sound to locate and identify objects, particularly used by animals like dolphins and bats. According to (Szabo, 2014), bats emit ultrasound pulses, ranging between 20-150 kHz, to catch an insect prey with a resolution of 2-15 mm. This involves signal processing such as:

1. Doppler shift (the relative speed of the target),

$$\Delta f = f_D - f_0 = f_0 \frac{c_s}{c_0} \cos(\theta) \quad (5.1)$$

2. time delay (distance to the target), and
3. frequency and amplitude in relation to distance (target object size and type recognition);

where the Doppler shift (or effect) is the perceived change in frequency (Doppler frequency  $f_D$  minus transmitted frequency  $f_0$ ) as a sound source with velocity  $c_s$  moves toward or away from the listener/observer with velocity  $c_o$  and angle  $\theta$ .

### 5.3.2 Staged classification and reconstruction pipeline

As depicted in Fig. 5.3, we take a staged approach to enhance scene and object reconstruction using audio-visual data. Our echoreconstruction prototype consists of two smartphones - one recording (top) and one emitting/reconstructing (bottom). Each audio emission is 100 ms of sound followed by 900 ms of silence to allow for the receiving microphone to capture reflections and reverberations (Section 5.3.3). After the 3D scan is complete, an .obj file containing geometry and texture information is generated. 1 second frames are extracted from the recorded video to generate audio and visual input into the EchoCNN neural networks (Section 5.3.4). These networks are independently trained to detect whether a surface is open or closed, estimate depth to the surface from the sound source, and classify the material of the surface. Using mobile accelerometer data and coarse audio with fine visual data to augment depth estimation will be explored as future work.

### 5.3.3 Sound source

A smartphone emits recordings of human experimenter voice, whistle, hand clap, pure tones (ranging from 63 Hz to 16 kHz), chirps, and noise (white, pink, and brownian). All of which can be generated as either pulsed (PW) or continuous waves (CW). PW is preferred for theoretical and empirical reasons. First, the transmission frequency  $f_0$  may experience considerable downshift as a result of absorption and diffraction effects (Szabo, 2014). Therefore, using pulsed waves independent for each emission is theoretically

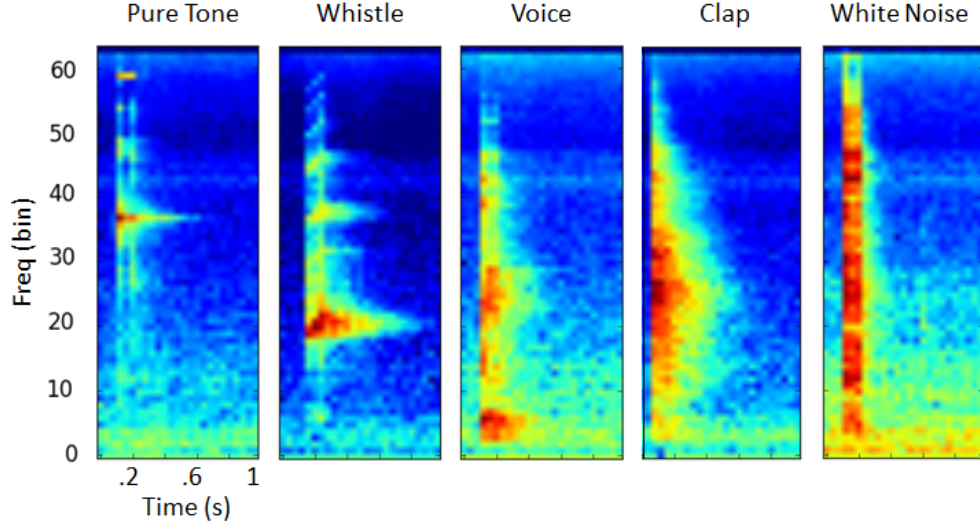


Figure 5.4: Mel-scaled spectrograms of recorded impulses of different sound sources used. *From left to right:* narrow to disperse spectra. Not shown are other pure tone frequencies, chirp, pink noise, and brownian noise. Horizontal axis is time and vertical axis is frequency.

better than continuous waves compared to  $f_0$ . Furthermore, Section 6.5 shows superior PW results over CW for the given classification tasks.

Pure tones were generated with default 0.8 out of 1 amplitudes using the Audacity computer program and center frequencies of 63 Hz, 125 Hz, 250 Hz, 500 Hz, 1 kHz, 2 kHz, 4 kHz, 8 kHz, and 16 kHz. Human voice ranges from about 63 Hz to 1 kHz (Long, 2014) (125 Hz to 8 kHz (Egan, 1988)) and an untrained whistler between 500 Hz to 5 kHz (Nilsson et al., 2008). Chirps were linearly interpolated from 440 Hz to 1320 Hz in 100 ms. A hand clap is an impulsive sound that yields a flat spectrum (Long, 2014). All sound sources were recorded and played back with max volume (Fig. 5.4). While recorded sounds were used for consistency, we plan to add live audio for augmentation and future ease of use during reconstruction. Please see our supplementary materials for spectrograms across all sound sources.

**Audio input:** audio was generated in pulsed waves (PW). One smartphone to emit the sound while performing a RGB-based reconstruction and the second smartphone to capture video. As future work, a single mobile device or Microsoft Kinect paired with audio chirps could be used for audio-visual capture and reconstruction instead of two separate devices. Each pulsed wave emitted into the scene was a total of 1 second consisting of an 100 ms impulse followed by silence. 1 second audio frames is based on the

Total room absorption $a$ using $a = \sum S\alpha$ at 250 Hz			
Real bathroom scene	S	$\alpha$	$a$ (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
Total $a$ =			69.23 sabins

Table 5.2: According to the Sabine Formula (Eq. 5.3.3), 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 ( $\text{ft}^2$ ), and  $\alpha$  is the sound absorption coefficient at a given frequency (decimal percent). At 250 Hz, the total room absorption  $a$  for our real-world bathroom scene is 69.23 sabins.

Sabine Formula of reverberation time for a compact room of like dimensions calculated as:

$$T = 0.05 \frac{V}{a} = 0.05 \frac{V}{\sum S\alpha} = (0.05 \frac{\text{sec}}{\text{ft}}) \frac{1,296 \text{ ft}^3}{69.23 \text{ ft}^2} = 0.94 \text{ sec} \quad (5.2)$$

where  $T$  is the reverberation time (time required for sound to decay 60 dB after source has stopped),  $V$  is room volume ( $\text{ft}^3$ ), and  $a$  is the total room absorption at a given frequency (e.g. 250 Hz). For the bathroom scene,  $V = 9 \text{ ft} * 16 \text{ ft} * 9 \text{ ft} = 1,296 \text{ ft}^3$  and  $a = 69.23 \text{ ft}^2$ , which is the sum of sound absorption from the materials in Table 5.2.

**Visual input:** images were captured from the same smartphone video as the audio recordings. Each corresponding image was cropped and grayscaled for illumination invariance and data augmentation. Image dimensions were 64 by 25 pixels. Visual data served as inputs for visual only and audio-visual model variation EchoCNN-AV.

#### 5.3.4 Model Architecture

To augment visually based approaches, we use a multimodal CNN with mel-scaled spectrogram and image inputs. First, we perform surface detection to determine if a space with depth jumps and holes is in error or in fact open (i.e. open/closed classification). In the event of error, we estimate distance from recorder to surface using audio-visual data for depth filtering and inpainting. Finally, we determine the material. All of these classifications are performed using our audio and audio-visual convolutional neural networks, referred to as EchoCNN-A and EchoCNN-AV (Fig. 5.3).

**Audio sub-network:** our frame-based EchoCNN-A consists of a single convolutional layer followed by two dense layers with feature normalization. Sampled at  $F_s = 44.1$  kHz to cover the full audible range, audio frames are 1 second mel-scaled spectrograms with STFT coefficients  $\chi$  (Eq. 6.3.2). Each audio example is classified independently and 1 second intervals to reflect an estimated reverberation time based on a compact room size (Eq. 5.3.3). With a 2048 sample Hann window ( $N$ ), 25% overlap, and hop length ( $H = 2048/4$ ), this results in a frequency dimension of 21.5 Hz (Eqn. 5.3.4) and temporal dimension of 12 ms (Eqn. 5.3.4) or 12% of each 100 ms pulsed audio. Each spectrogram is individually normalized and downsampled to a size of 62 frequency bins by 25 time bins.

We define the frequency spectral coefficients (Miller, 2015) as:

$$\chi(m, k) = \sum_{n=0}^{N-1} x(n + mH)w(n)\exp(-2\pi i k n/N) \quad (5.3)$$

for  $m^{th}$  time frame and  $k^{th}$  Fourier coefficient with real-valued DT signal  $x : \mathbb{Z} \rightarrow \mathbb{R}$ , sampled window function  $w(n)$  for  $n \in [0 : N - 1] \rightarrow \mathbb{R}$  of length  $N \in \mathbb{N}$ , and hop size  $H \in \mathbb{N}$  (Miller, 2015).  $\mathbb{R}$  denotes continuous time and  $\mathbb{Z}$  denotes discrete time. Equal to  $|\chi(m, k)|^2$ , spectrograms have been demonstrated to perform well as inputs into convolutional neural networks (CNNs) (Huzaifah, 2017b). Their horizontal axis is time and vertical axis is frequency.

$$F_{coef}(k) = \frac{k\dot{F}_s}{N} = k \frac{44100}{2048} = k * 21.5 \text{ Hz} \quad (5.4)$$

$$T_{coef}(m) = \frac{m\dot{H}}{F_s} = m \frac{2048 * 0.25}{44100} = m * 0.012 \text{ seconds} \quad (5.5)$$

A hop length of  $H = N/2$  achieves a reasonable temporal resolution and data volume of generated spectral coefficients (Miller, 2015). Temporal resolution is important in order to detect when a reflecting sound reaches the receiver. Therefore, we decided to use a shorter window length  $N = 2048$  instead of  $N = 4096$  for instance. This resulted in a shorter hop length and accepting the trade-off of a higher temporal dimension for increased data volume.

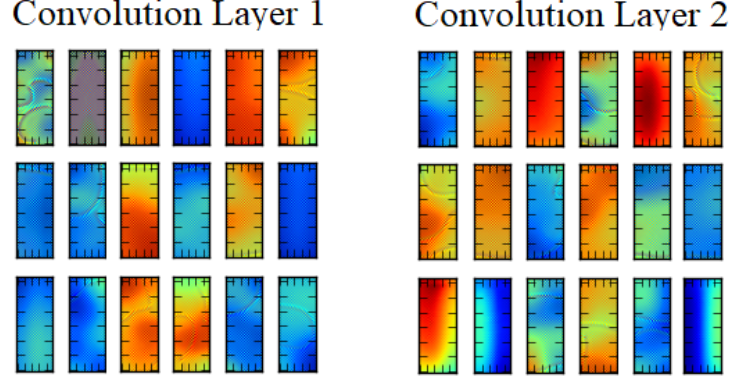


Figure 5.5: Sample visualizations of the filters for the two convolutional layers in the audio-based EchoCNN-A neural network. The model learns filters for octave bands, frequencies, reflections, reverberations, and damping.

**Visual sub-network:** while audio information is generally useful for all three classifications tasks (Table 3.2) visual information is particularly useful to aid material classification. We use ImageNet (Krizhevsky et al., 2012b) as a visual-based baseline to compare to our audio and audio-visual methods. It also serves as an input into our audio-visual merge layer. Future work will explore whether or not another image classification method is better suited as a baseline and to fuse with audio.

**Merge layer:** we evaluated concatenation and multi-modal factorized bilinear (MFB) pooling (Yu et al., 2017a) to fuse audio and visual fully connected layers. Concatenation of the two vectors serves as a straightforward baseline. MFB allows for additional learning in the form of a weighted projection matrix factorized into two low-rank matrices.

$$z_i = x^T W_i y = x^T U_i V_i^T y = 1^T (U_i^T x \circ V_i^T y) \quad (5.6)$$

where  $k$  is the factor or latent dimensionality with index  $i$  of the factorized matrices,  $\circ$  is the Hadamard product or element-wise multiplication, and  $1 \in \mathbb{R}^k$  is an all-one vector.

### 5.3.5 Loss Function

Categorical cross entropy loss is used for EchoCNN inferences. For open/closed predictions, categorical cross entropy loss is used instead of binary if estimating the extent of the surface opening (e.g. all the way open, halfway open, or closed). A regression model is not used for depth estimation because ground



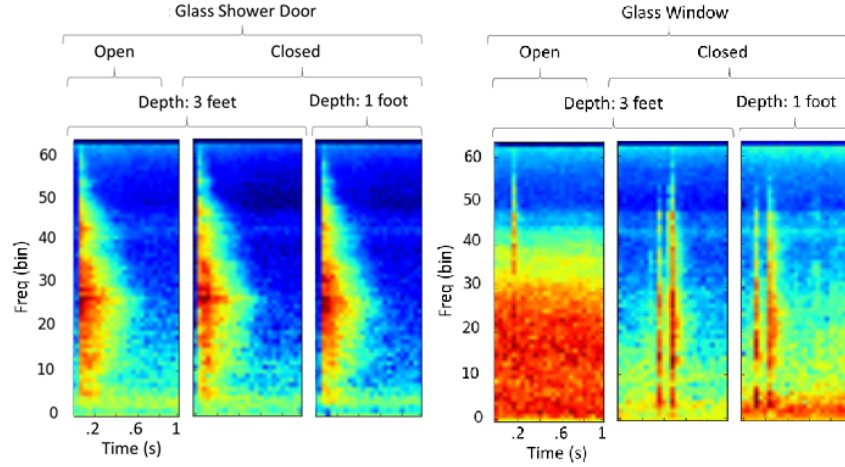


Figure 5.6: Spectrograms from a recorded hand clap in front of an interior glass shower door and exterior glass window. For the interior door, reflected sounds experience intensified damping as we go from opened (*left*) to closed (*middle*) and then from 3 feet to 1 foot depth (*right*). Damping increases with fewer late reverberations and intensity increases with more early reflections. For the exterior window, closing it decreases outside noise up to a distance.

truth data is collected in 1 foot increments within the free field for better noise reduction (Egan, 1988). The Softmax function is used for output activations.

### 5.3.6 Depth filtering and planar inpainting

The outputs of our EchoCNN inform enhancements for 3D reconstruction (Algorithm ??). If depth jumps in the reconstruction are first classified as an open surface, then no change is required other than filtering loose geometry and small components. Otherwise, there is a planar discontinuity (e.g. window or mirror) that needs to be filled. With depth estimated by EchoCNN, we filter the initial 3D mesh to within a threshold of that depth. This gives us the plane size needed to fill. Finally, EchoCNN classifies its surface material.

## 5.4 Datasets

Our audio-based EchoCNN-A and audio-visual EchoCNN-AV convolutional neural networks are trained across nine octave bands with center frequencies 63 Hz, 125 Hz, 250 Hz, 500 Hz, 1 kHz, 2 kHz, 4 kHz, 8 kHz, and 16 kHz. Training is done using these pulsed pure tone impulses along with experimenter hand clap. The hold out test data is comprised of sound sources excluded from training - white noise,

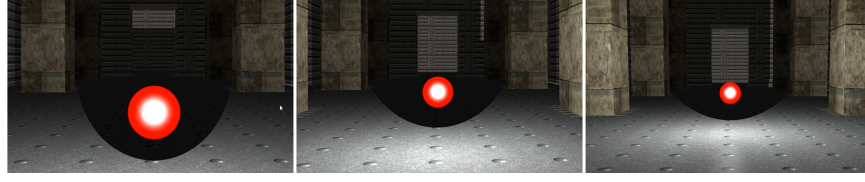


Figure 5.7: Listener at different distances of 1, 2, 3 ft from sound source (red dot) in a virtual environment used to generate synthetic audio-visual data. In addition to open/closed, depth, and material, we make synthetic, unmixed reflection separation data (direct, early, or late) available for future research.

experimenter whistle, and voice. The test set contains sound sources not in the training set to evaluate generalization.

#### 5.4.1 Real and synthetic datasets

**Real:** training data is comprised of 1 second pulsed spectrograms (Fig. 5.6) from recorded pure tones, experimenter hand claps, brownian noise, and pink noise (N=857). Training and test examples were collected via video recordings and labeled for material, open/closed, and in 1 ft depth increments based on the distance from the surface. Nine octaves of pure tones, hand claps, and white noise cover a disperse range of frequencies and were thus used to train our models.

The hold out test dataset consists of 1 second pulsed spectrograms from recorded experimenter voice, whistle, chirp, and white noise (N=227). Voice and whistle recordings were chosen for the hold out test set to ease future transition to live and hands-free emitted sounds during reconstruction. Hold out test data is excluded from training and only evaluated during testing. While the same hold out sets were used for visual and audio-visual evaluation, unheard is not the same as unseen. Unheard audio can have the same visual appearance between training and test. Other new training and test datasets for visual and audio-visual methods will be future work.

**Synthetic:** we employ a ray-based geometric sound propagation approach (Schissler and Manocha, 2011). Given scene materials (e.g. carpet, glass, painted, tile, etc.), a sound source (e.g. voice), 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:

$$S_f(t) = \sum \delta(t - t_j) I_{j,f} \quad (5.7)$$

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 these sound rays collide in the scene, their paths change based on absorption and scattering coefficients of the colliding objects. Common acoustic material properties can be referenced in (Egan, 1988). 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 corresponding to 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:

$$w_{f,m} = \frac{\sum I_{j,f} d_{j,m}}{\sum I_{j,f}} \quad (5.8)$$

where  $d_{j,m}$  is the number of times rays on path  $j$  collide with material  $m$ , weighted according to the sound intensity  $I_{j,f}$  of the path  $j$ . To mirror our real-world data, sound source directivity was disabled. Future work is needed to compare ambient and directed sound sources. This data may also be used for material sound separation.

## 5.5 Experiments and Results

Overall, 71.2% of hold out reflecting sounds and 100% of audio-visual frames were correctly classified as an open or closed boundary in the home (Table 3.2). 71.8% of 1 second audio frames were correctly classified as 1 ft, 2 ft, or 3 ft away from the surface based on audio alone; 89.5% when concatenating with its corresponding image. Finally, 77.4% of audio and 100% of audio-visual inputs correctly labeled the surface material.

ImageNet, a visual only baseline, is higher at 78.1% than audio-only EchoCNN-A for open/closed classification. This is partly due to the fact that the hold out set was to test audio generalization (i.e. unheard sound sources). But unheard sound sources does not guarantee unseen visual data. Images similar to those found in training are present in test. A proper hold out set based on image (e.g. different depths) should be evaluated as future work.

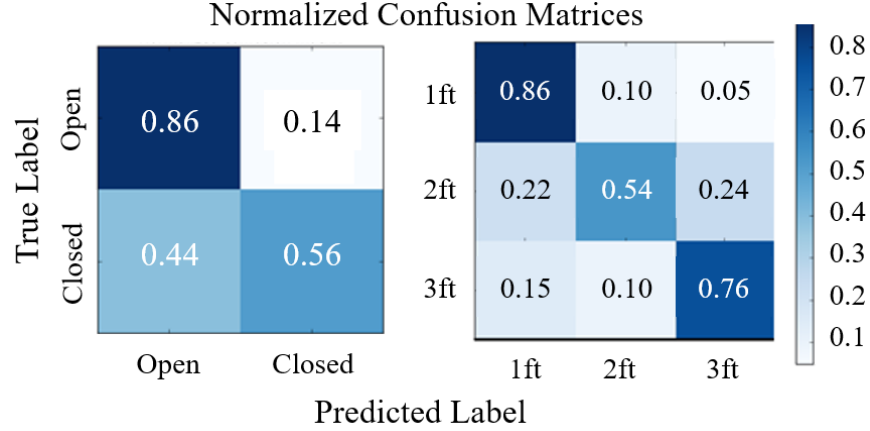


Figure 5.8: EchoCNN-A (*Left*) Confusion matrix to classify open/closed for an interior glass shower door. Open predictions (86%) were more accurate than closed (56%). (*Right*) Confusion matrix to classify depth from same interior glass door. Notice that our EchoCNN is learning to differentiate distance based on reflecting sounds from pulsed ambient waves of a smartphone.

### 5.5.1 Experimental setup

Listener (top smartphone, e.g. Galaxy Note 4) and sound source (bottom smartphone, e.g. iPhone 6) are separated vertically by 7 cm. Pulsed sounds are emitted 3 feet, 2 feet, and 1 foot away from the reconstructing surface. Three feet was selected to remain in the free field. Beyond that, there will be less noise reduction due to reflecting sounds in the reverberant field (Egan, 1988). Within a few feet of the reconstructing surface also create finer detail reconstructions.

We labeled our data based on scene, sound source, and surface properties - type of surface, material, and depth from sound source. The training set included pulsed sounds of pure tone frequencies, a single hand clap, brownian noise, and pink noise. The hold out test set consisted of voice, whistle, chirp, and white noise. For rooms with different sound-absorbing treatments, our real-world recordings include a bedroom (e.g. carpet and painted) and bathroom (e.g. tiled).

### 5.5.2 Implementation details

We implemented all EchoCNN and baseline models with Tensorflow (Abadi et al., 2015) and Keras (Chollet et al., 2015). Training was performed using a TITAN X GPU running on Ubuntu 16.04.5 LTS. We used categorical cross entropy loss with Stochastic Gradient Descent optimized by ADAM (Kingma and Ba, 2014). Using a batch size of 32, remaining hyperparameters were tuned manually based on a separate validation set. We make our real-world and synthetic datasets available to aid future research in this area.

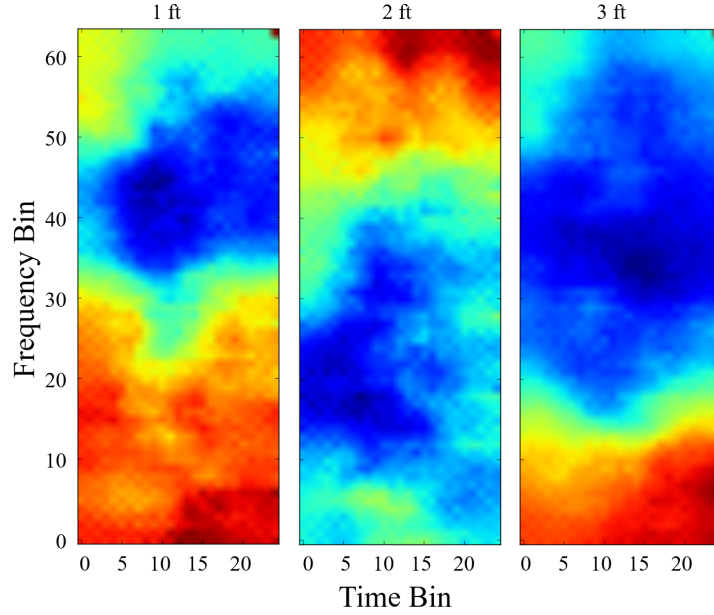


Figure 5.9: *From left to right:* audio input (i.e. mel-scaled spectrogram) which would produce the highest activation for a given depth class from 1 ft, 2 ft, and 3 ft away from an object. Longer reverberation times tend to occur at lower frequencies (3 ft) than at high frequencies (1 and 2 ft) due to typical high frequency damping and absorption.

#### 5.5.2.1 Initial 3D Reconstruction

We evaluated the following smartphone-based reconstruction applications to obtain an initial 3D geometry for which our method would enhance. The Astrivis application, based on (Tanskanen et al., 2013), generates better live 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.

#### 5.5.3 Results by source frequency and object size

We will evaluate dynamic source frequencies based on the physical size of the objects, since sound wave behavior relates to wavelength. For example, if an object is much smaller than the wavelength, the sound flows around it rather than scattering (Long, 2014).

$$\lambda = \frac{c}{f} \quad (5.9)$$

where  $\lambda$  is wavelength (ft) of sound in air at a specific frequency,  $f$  is frequency (1 Hz), and  $c$  is speed of sound in air (ft/s).

#### 5.5.4 Activation Maximization

The objective of activation maximization is to generate an input that maximizes layer activations for a given class. This provides insights into the types of patterns the neural network is learning. Fig. 5.9 shows the different inputs that would maximize EchoCNN activations for depth estimation. Notice lower frequencies tend to occur at 3 ft (longer reverberation times) than at 1 and 2 ft (high frequencies) due to typical high frequency damping and absorption.

#### 5.5.5 Applications

When using a head mounted display (HMD) users are alerted within the virtual environment, when they approach the physical space boundaries established during room setup. 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. Depth estimation from audio can also be used to unmix and place unseen but heard sound sources from video into a virtual environment. In addition to scene reconstruction, echoreconstruction also reconstructs audio (Fig. 5.12).

### 5.6 Conclusion and Future Work

To the best of our knowledge, these are the first audio and audio-visual techniques introduced for enhancing scene reconstructions that contain windows and mirrors. Our multi-smartphone prototype and staged echoreconstruction pipeline emits and receives pulsed audio from a variety of sound sources for surface detection, depth estimation, and material classification. These classifications enhance scene and object 3D reconstruction by resolving planar discontinuities caused by open spaces and reflective surfaces using depth filtering and planar filling. Our prototype performs well compared to baseline methods given our experiment results for multiple real-world and virtual scenes containing windows, mirrors, and open surfaces. We make publicly available our real and synthetic audio-visual ground truth data in addition to reflection separation data (direct, early, or late reverberations) for future research.



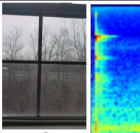
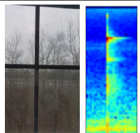
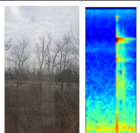

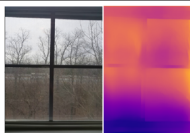
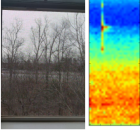
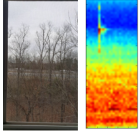
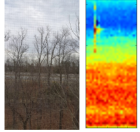

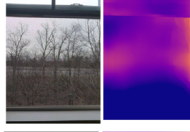

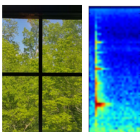
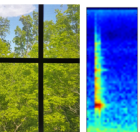
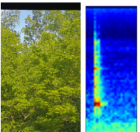

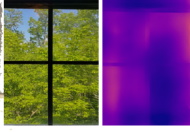
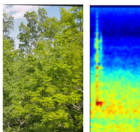
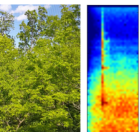
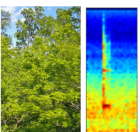

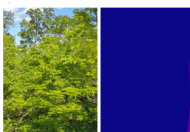


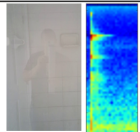
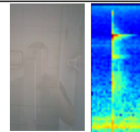
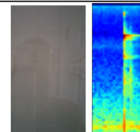

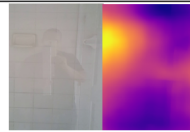
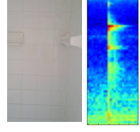
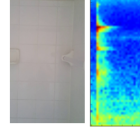
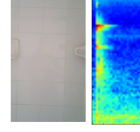

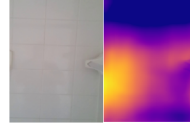
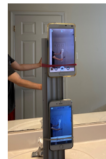
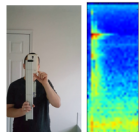
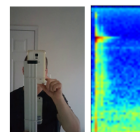
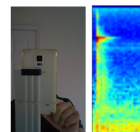
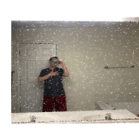
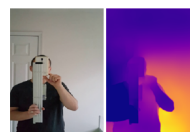

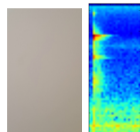
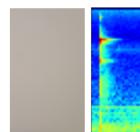
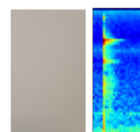
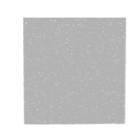

Scene	Est. Reverb Time (250 Hz)	Experimental Setup	Object	3 ft	2 ft	1 ft	Echo-Reconstruction 3D Point Cloud	Prior Work: High Quality Monocular Depth Est. via Transfer Learning
Bedroom 	0.77 sec		Closed Window (Winter) 8 kHz					
			Open Window (Winter) 8 kHz					
			Closed Window (Spring) 1 kHz					
			Open Window (Spring) 1 kHz					
Bathroom 	0.94 sec		Closed Shower Door					
			Open Shower Door					
			Large Mirror					
			Painted Wall					

Figure 5.10: We evaluated our method on real and virtual scenes. *Column 1*: we used the off-the-shelf MagicPlan app to obtain 3D models and dimensions to calculate estimated reverberation time based on room size and materials. Our experimental setups consists of a two smartphone prototype. One phone performs an initial reconstruction using state-of-the-art commercial Astrivis application and also emits pulsed audio. The second phone captures video for audio-visual input data into our EchoCNN. We tested glass, mirror, and other objects and surfaces within each scene at different depths, materials, and open/closed. Using audio, we noticed noise reduction between winter and spring due to more foliage on the trees. We also observed flutter echoes, which can be heard as a "rattle" or "clicking" from a hand clap and have been simulated in spatial audio (Halmrast, 2019). They became more pronounced the closer to the wall surface in the bathroom scene. Background UV textures are placed at a fixed 1 ft (0.3 m) behind estimated surface depth. Audio unable to augment failure cases of the shower from initial RGB-based reconstructions using either (Tanskanen et al., 2013) or (Metashape, 2020). We leave calculating the background depth as future work. We compare our 3D reconstructions to depth estimates based on related work.



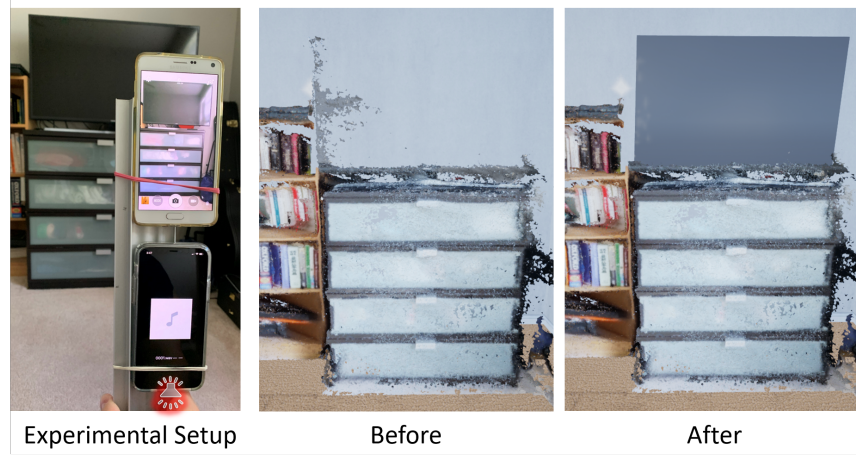


Figure 5.11: Echoreconstruction of a TV on a dresser. (From left to right) Photo of prototype system in action, initial 3D reconstruction, depth filtering applied, and resulting echoreconstruction. Semantic rendering is applied post-processing during the render stage of the pipeline.

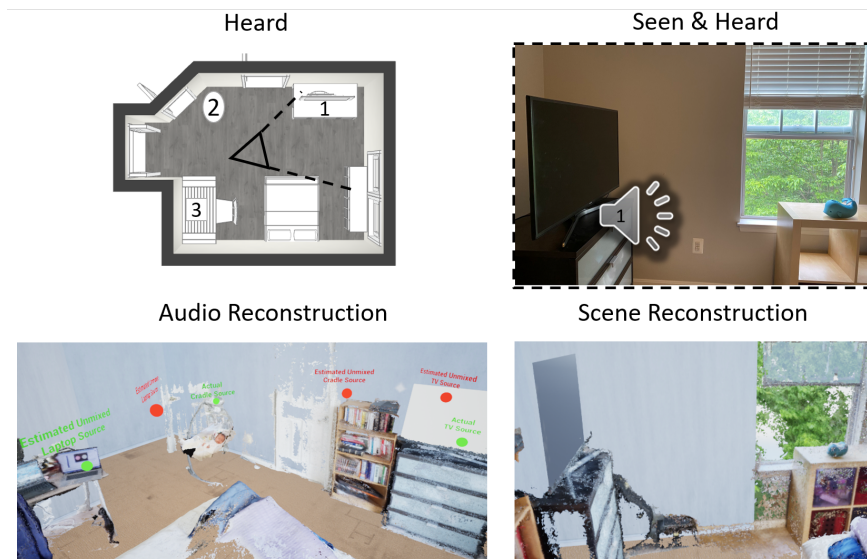


Figure 5.12: EchoCNN may also be used to reconstruct the audio of a scene from video. Instead of depth estimation, our method can be trained to approximate sound source position, which is especially useful for objects that are outside of the camera field of view. Ground truth (green dots) and estimated (red dots) sound source placements. Seen and heard sound source (TV) from the video capture placed more accurately than unseen but heard sound sources (cradle and laptop). Audio-visual compared to audio only. Please see our supplementary video for a VR demo and improved sound source placement as future work.



**Future Work:** To further extend this research, performing audio emission, reception, and 3D reconstruction simultaneously and in real-time instead of having a staged approach would be one possible alternative to explore. This approach could possibly enable mapping classifications to 3D geometry more densely than fusing RGB-D, tracking, or Iterative Closest Point (ICP) (Izadi et al., 2011). An integrated approach may not only be more efficient but also more effective by using audio feedback as part of the reconstruction code. Another possible avenue of exploration is to investigate the impact of live audio for training and/or testing our neural network variations. With a defined set of output classes for EchoCNN, alternative baselines such as Non-Negative Matrix Factorization (NMF), source separation techniques, and the pYIN algorithm (Mauch and Dixon, 2014) to extract the fundamental frequency  $f_0$ , i.e. the frequency of the lowest partial of the sound, are suggested as future work. Finally, our current implementation holds out voice and whistle data, which is different from the audio used during training. However, unheard sounds does not equate to unseen images. Therefore, some insights can be possibly gained by experimenting with a different training dataset for testing audio-only, visual-only, and audio-visual methods.

## CHAPTER 6: AUDIO-VISUAL OBJECT RECONSTRUCTION FROM VIDEO<sup>1</sup>

This chapter describes a multimodal single and multi-frame neural network for 3D reconstructions using audio-visual inputs. Our trained reconstruction LSTM autoencoder 3D-MOV accepts multiple inputs to account for a variety of surface types and views. To the best of our knowledge, our single and multi-frame model is the first audio-visual reconstruction neural network for 3D geometry and material representation.

### 6.1 Introduction

Deep neural networks trained on single- or multi-view images have enabled 3D reconstruction of objects and scenes using RGB and RGBD approaches. These models generate 3D geometry volumetrically (Boscaini et al., 2016; Choy et al., 2016; Xie et al., 2018a) and in the form of point clouds (Han et al., 2019; Qi et al., 2016a, 2017a). With these reconstructions, additional networks have been developed to use the 3D geometry as inputs for object detection, classification, and segmentation in 3D environments (Atzmon et al., 2018; Qi et al., 2017b). However, existing methods still encounter a few challenging scenarios for 3D shape reconstruction (Boscaini et al., 2016).

One such challenge is occlusion in cluttered environments with multiple objects in a scene. Another is spatial resolution. Volumetric methods such as voxelized reconstructions (Maturana and Scherer, 2015) are primarily limited by resolution. Point cloud representations of shape avoid issues of grid resolution, but instead need to cope with issues of point set size and approximations. Existing methods also are challenged by transparent and highly reflective or textured surfaces. Self-occlusions and occlusions from other objects can also hinder image-based networks, necessitating the possible adoption of multimodal neural networks.

To address these limitations, we propose to use audio-visual input for 3D shape and material reconstruction. A single view of an object is insufficient for 3D reconstruction as only one projection of the object can be seen, while multi-view input does not intrinsically model the spatial relationships between

---

<sup>1</sup> This chapter has been prepared for conference submission.

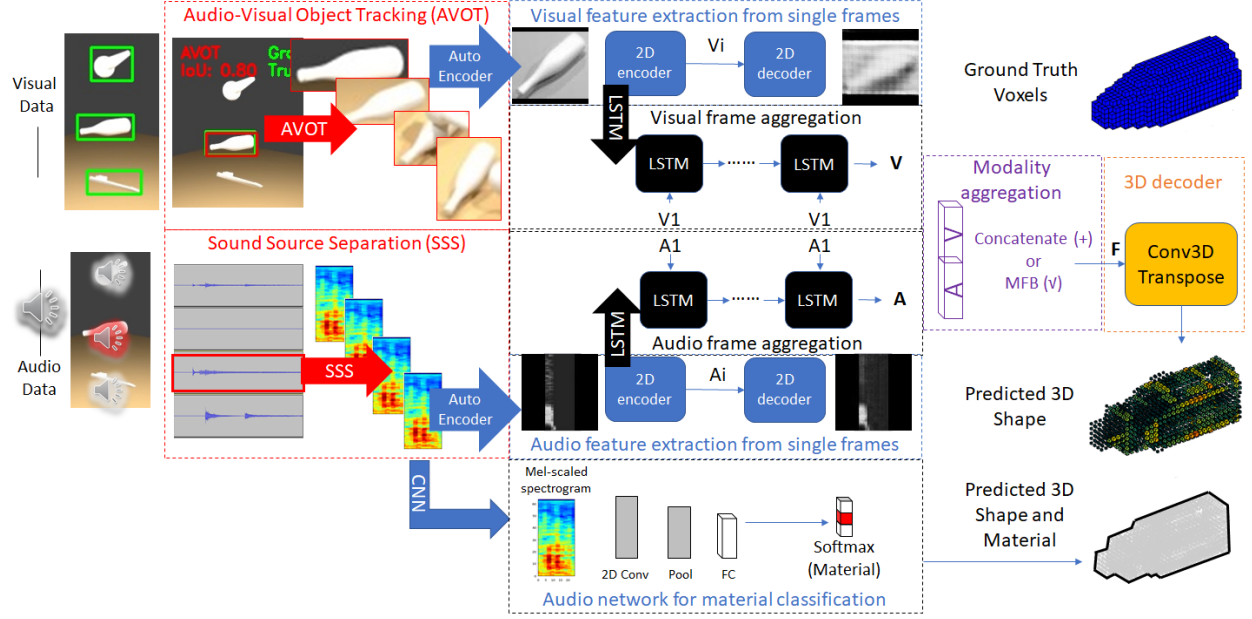


Figure 6.1: Our 3D-MOV neural network is a multimodal LSTM autoencoder optimized for 3D reconstructions of single ShapeNet objects and multiple objects from Sound20K video. During training, a LSTM autoencoder is trained to reconstruct 2D image and spectrogram inputs. 3D shape reconstructions are then generated by fine tuning the fused encodings of each modality for 3D voxel output. The network has recurrent LSTM layers for temporal consistency. Adding audio enhances learning for object tracking, material classification, and reconstruction when multiple objects collide, self-occlude, or are transparent.

views. By providing a temporal sequence of video frames, we strengthen the relationships between views, aiding reconstruction. We also include audio as an input, in particular, *impact sounds* resulting from interactions between the object to be reconstructed and the surrounding environment. Impact sounds provide information about the material and internal structure of an object, providing complementary cues to the object’s visual appearance. We choose to represent our final 3D shape using voxel representation due to their state-of-the-art performance in classification tasks. To the best of our knowledge, our audio-visual network is the first to reconstruct multiple 3D objects from a single video.

**Main Results:** In this paper, we introduce a new method to reconstruct high-quality 3D objects from a sequence of images and sound, the main contributions of this work can be summarized as follows.

- A multimodal LSTM autoencoder neural network for geometry and material reconstruction from audio and visual data is introduced;
- The resulting implementation has been tested on voxel, audio, and image datasets of objects over a range of different geometries and materials;

- Experimental results of our approach demonstrate the reconstruction of single sounding objects and multiple colliding objects in a virtual scene;
- Audio-augmented datasets with ground truth object tracking bounding boxes are made available for future research.

## 6.2 Related Work

Computer vision research continues to push state-of-the-art reconstruction and segmentation of objects in a scene (Dai et al., 2017c). However, there still remain research opportunities in 3D reconstruction. Wide baselines limit the accuracy of feature correspondences between views. Challenging objects for reconstruction include thin or small objects (e.g. table legs), and classes of objects that are transparent, occluded, or have much higher shape variation than other classes (e.g. lamps, benches, and tables compared to cabinets, cars, and speakers for example). In this section, we review previous work relating to 3D reconstruction, multimodal neural networks, and reconstruction network structures.

### 6.2.1 3D Reconstruction

Deep learning techniques have produced state-of-the-art 3D scene and object reconstructions. These models take an image or series of images and generate a reconstructed output shape. Some methods produce a transformed image of the input, intrinsically representing the 3D object structure (Odena et al., 2017; hong Tsai, 2018; Mao et al., 2017; Mirza and Osindero, 2014; Lun et al., 2017). 3D voxel grids provide a shape representation which is easy to visualize and works well with convolution operations (Choy et al., 2016; Girdhar et al., 2016; Riegler et al., 2017; Qi et al., 2016b; Hu et al., 2018; Wu et al., 2016). In more recent work, point clouds have also been found to be a viable shape representation for reconstructed objects (Hedman et al., 2017; Fan et al., 2017).

### 6.2.2 Multimodal Neural Networks

Neural networks with multiple modalities of inputs help cover a broader range of experimental setups and environments. Common examples include visual question answering (Cadene et al., 2019), vision and touch (Lee et al., 2019), and other multisensory interactions (Klemen and Chambers, 2012). Multiple

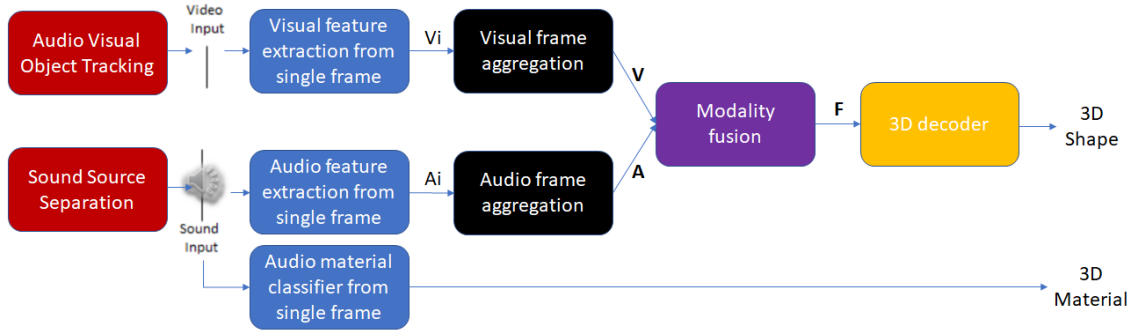


Figure 6.2: We first separate audio-visual data using object tracking (Section 6.3.1) and sound source separation (Section 6.3.2). Features from audio and visual subnetworks for each object are aggregated by LSTM autoencoders and then fused using addition, concatenation, or a bilinear model (Yu and Tao, 2014). Finally, 3D geometry is reconstructed by a 3D decoder and audio classified material applied to all voxels.

modes may also take the form of image-to-image translation, e.g. domain transfer (Huang et al., 2018).

Using local and global cropped parts of the images (i.e. bounding boxes) have also been shown to serve as a mode of context to supervise learning (Reed et al., 2016).

Audio-visual specific multimodal neural networks have also proven effective for speech separation (Ephrat et al., 2018b) as well as sound localization (Zhao et al., 2018; Owens and Efros, 2018; Konno et al., 2020; Arandjelović and Zisserman, 2017). Audio synthesis conditioned on images is also enabled as a result of these combined audio-visual datasets (Zhou et al., 2018). Please see a survey and taxonomy on multimodal machine learning (Baltrusaitis et al., 2017) and multimodal deep learning (Ngiam et al., 2011a) for more information.

### 6.2.3 Reconstruction Network Structures

While single view networks perform relatively well for most object classes, objects with concave structures or classes of objects with large variations in shape tend to require more views. 3D-R2N2 (Choy et al., 2016) allows for both single and multi-view implementations given a single network. Other recurrent models include learning in video sequences (Chong and Tay, 2017; Hasan et al., 2016), Point Set Generation (Fan et al., 2017), and Pixel Recurrent Neural Network (PixelRNN) (van den Oord et al., 2016c). Methods have also been developed to ensure temporal consistency (Xie et al., 2018b) and use generative techniques (Gwak et al., 2017). T-L network (Girdhar et al., 2016) and 3D-R2N2 (Choy et al., 2016) are most similar to our 3D-MOV reconstruction neural network. Building on these related works, we fuse audio as an additional input and temporal consistency in the form of LSTM layers (Fig. 6.2).

### 6.3 Technical Approach

In this work, we reconstruct the 3D shape and material of sounding objects given images and impact sounds. Using audio and visual information, we present a method for reconstruction of single instance ModelNet objects augmented with audio and multiple objects colliding in a Sound20K scene from video. In this section, we cover visual representations from object tracking (Section 6.3.1) and audio obtained from sound source separation of impact sounds (Section 6.3.2) that serve as inputs into our 3D-MOV reconstruction network (Section 6.4).

#### 6.3.1 Object Tracking and Visual Representation

Since an entire video frame may contain too much background, we use object tracking to track and segment different objects. This tracking is performed using the Audio-Visual Object Tracker (AVOT) (Wilson and Lin, 2020a). Similar to the Single Shot MultiBox Detector (SSD) (Liu et al., 2015), AVOT is a feed-forward convolutional neural network that classifies and scales a fixed number of anchor bounding boxes to track objects in a video. While 3D-MOV aggregates audio-visual features before decoding, AVOT fuses audio-visual inputs before its base network. With additional information from audio, AVOT defines an object based on both its geometry and material.

We use AVOT over other algorithms, such as YOLO (Redmon et al., 2015b) or Faster R-CNN (Ren et al., 2015b), because of the availability of audio and need for higher object-tracking accuracy given occlusions caused by multiple objects colliding. Unlike CSR-DCF (Lukezic et al., 2016), AVOT automatically detects objects in the video without initial markup of bounding boxes. For future work, a scheduler network or a combination of object trackers is worth considering as well as use of Common Objects in Context (COCO) (Lin et al., 2014a) and SUN RGB-D (Song et al., 2015; Silberman et al., 2012a; Janoch et al., 2011; Xiao et al., 2013) datasets for initialization and transfer learning.

The output from tracking is a series of segmented image frames for each object, consisting of the contents of its tracked bounding box throughout the video. These segmented frames are grayscaled and resized to a consistent input size of 88 by 88 pixels. While resizing, we maintain aspect ratio and pad to square the image. These dimensions were automatically chosen to account for the size of objects in our Sound20K dataset and to capture their semantic information. Scenes included one, two, and three colliding objects with materials such as granite, slate, oak, and marble. For our single-frame, single impact sound

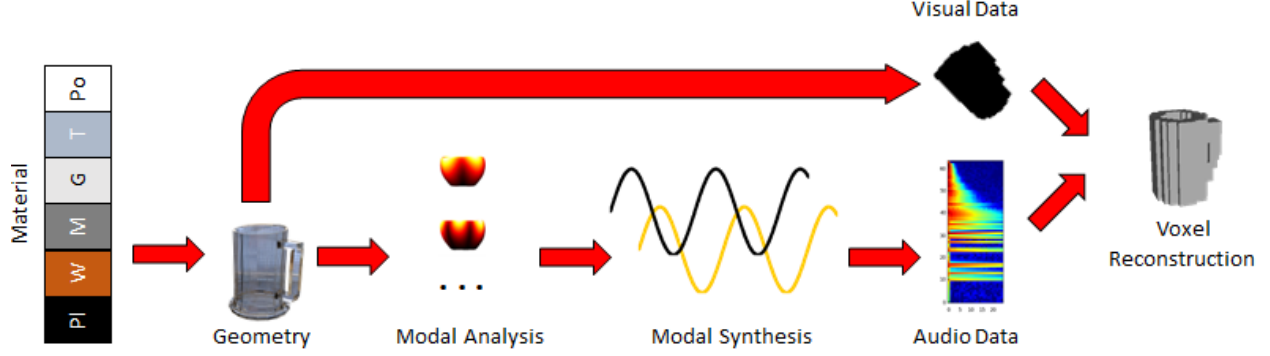


Figure 6.3: For our single impact sound analysis using ShapeNet, we build multimodal datasets using modal sound synthesis to produce spectrograms for audio input and images of voxelized objects as an estimate of shape. Please note that audio used from ISNN (Auston Sterling and Lin, 2018) was generated for voxelized models as a result of the sound synthesis pipeline requiring watertight meshes. Unmixed Sound20K audio was available from the generated synthetic videos.

evaluations, we resized ShapeNet’s 224 x 224 image size. For comparison, other image sizes from related work include MNIST, 28 x 28; 3D-R2N2, 127 x 127; ImageNet, 256 x 256.

### 6.3.2 Sound Source Separation of Impact Sounds and Audio Representation

For single frame reconstruction, we synthesize impact sounds on ShapeNet (Auston Sterling and Lin, 2018), illustrated in Fig. 6.3. For multiple frames, we take as input a Sound20K video showing one or more objects moving around a scene. These objects strike one another or the environment, producing impact sounds, which can be heard in the audio track of the video. We refer to these objects, dynamically moving through the scene and generating sound due to impact and collision, as *sounding objects*. Sound20K provides mixed and unmixed audio which can be used directly or to train algorithms for sound source separation (Wang et al., 2014; Koretzky et al., 2017; Scallie et al., 2017). While prior work to localize objects using audio-visual data exists (Arandjelović and Zisserman, 2017; Zhao et al., 2018), automatically associating separated sounds with corresponding visual object tracks in the context of the reconstruction task remains an area of future work.

Initially, Sound20K and ShapeNet audio are available as time series data, sampled at 44.1 kHz to cover the full audible range. The audio is converted to mel-scaled spectrograms for neural network inputs, which effectively represent the spectral distribution of energy over time. Each spectrogram is 3 seconds for a single frame (ShapeNet) and 0.03 seconds per multi-frame (Sound20K) with an overlap of 25%. Audio spectrograms are aligned temporally with their corresponding image frames from video, forming the audio-

visual input for queries. They are generated with discrete short-time Fourier transforms (STFTs) using a Hann window function.

$$\chi(m, k) = \sum_{n=0}^{N-1} x(n + mH)w(n)\exp(-2\pi i k n / N) \quad (6.1)$$

for  $m^{th}$  time frame and  $k^{th}$  Fourier coefficient with real-valued DT signal  $x : \mathbb{Z} \rightarrow \mathbb{R}$ , sampled window function  $w(n)$  for  $n \in [0 : N - 1] \rightarrow \mathbb{R}$  of length  $N \in \mathbb{N}$ , and hop size  $H \in \mathbb{N}$  (Miller, 2015).

### 6.3.2.1 Single View, Single Impact Sound

Single view inputs are based on ShapeNet, a repository of 3D CAD models based on WordNet categories. Evaluations were performed on voxelized versions of ShapeNet’s (Chang et al., 2015), ModelNet10 and ModelNet40 models (Wu and Xiao, 2015), and image views of these datasets from 3D-R2N2 (Choy et al., 2016). To generate audio for these objects to be used for our multi-modal 3D-MOV neural network, we use data from Impact Sound Neural Network (Auston Sterling and Lin, 2018). This work synthesized impact sounds for voxelized ModelNet10 and ModelNet40 models (Wu and Xiao, 2015) using modal analysis and sound synthesis. Modal analysis is precomputed to obtain *modes* of vibration for each object and sound synthesized with an amplitude determined at run-time given the hit point location on the object and impulse force. The modes are represented as damped sinusoidal waves where each mode has the form

$$q_i = a_i e^{-d_i t} \sin(2\pi f_i t + \theta_i), \quad (6.2)$$

where  $f_i$  is the frequency of the mode,  $d_i$  is the damping coefficient,  $a_i$  is the excited amplitude, and  $\theta_i$  is the initial phase.

### 6.3.2.2 Multi-Frame, Multi-Impact

Multi-frame inputs to our system consist of Sound20K (Zhang et al., 2017d) videos that may contain multiple sounding objects, possibly of similar sizes, shapes, and/or materials. This synthetic video dataset contains audio and video data for multiple objects colliding in a scene. Sound20K consists of 20,378 videos generated by rigid-body simulation and impact sound synthesis pipeline (Doug L. James and Pai,



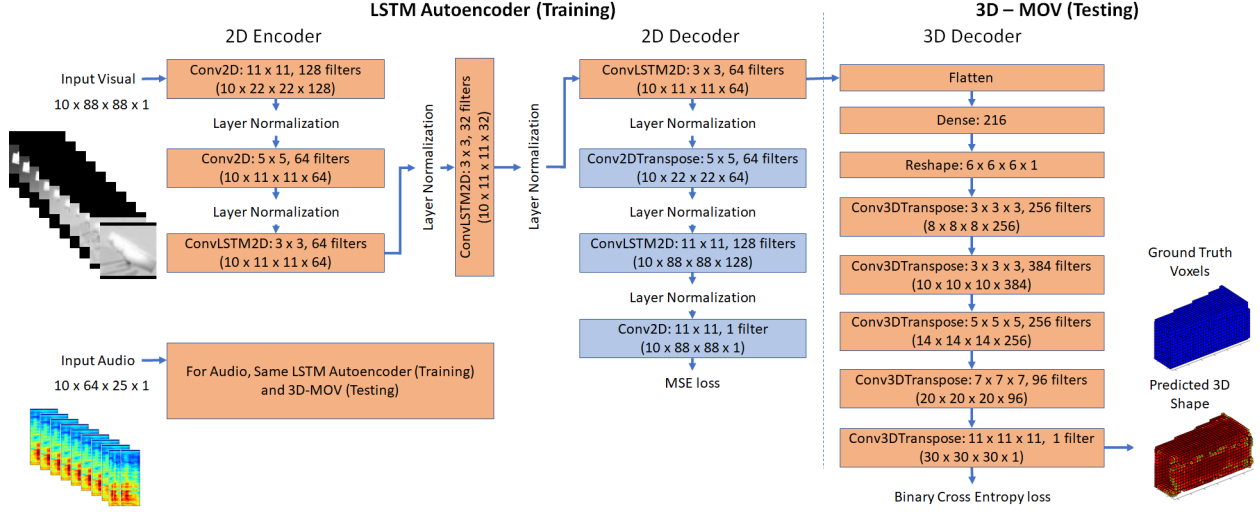


Figure 6.4: We separately train audio and visual autoencoders to learn encodings and fine-tune for our 3D reconstruction task. We replace the 2D decoder by a five deconvolutional layer 3D decoder to generate a  $30^3$  voxel grid. The separate audio-visual LSTM autoencoders are flattened and merged to form the dense layer. Here, the predicted 3D shape voxels are displayed based on a threshold of 0.3.

2006). Visually, Sound20K (Zhang et al., 2017d) objects can be separated from one another through tracking of bounding boxes. However, audio source separation can be more challenging, particularly for unknown objects. While Sound20K provides separate audio files for each object that can be used, the audio data can also be used to train sound source separation techniques (Wang et al., 2014; Koretzky et al., 2017; Scallie et al., 2017) to learn to unmix audio to individual objects by geometry and material. As future work, we will compare the impact on reconstruction quality and performance if we were to use combined, unmixed audio for each object. We will also compare impact of using source separated sounds versus ground truth unmixed audio.

## 6.4 3D-MOV Network Structure

Our 3D-MOV network is a multi-modal LSTM autoencoder optimized for 3D reconstructions of multiple objects from video. Like 3D-R2N2 (Choy et al., 2016), it is recurrent and generates a 3D voxel representation. However, to the best of our knowledge, our 3D-MOV network is the first audio-visual reconstruction network for 3D object reconstruction. After object tracking and sound source separation, we separately train autoencoders to extract visual and audio feature from each frame (Section 6.4.1). While the 2D encoder weights are reused, the 2D decoders are discarded (blue rectangles in Fig. 6.4) and re-

placed with 3D decoders for learning to reconstruct voxel outputs of the tracked objects based on given 2D images and spectrograms. Using a merge layer such as addition, concatenation, or a bilinear model (Yu and Tao, 2014), our method 3D-MOV fuses the results of the audio and visual subnetworks comprised of LSTM autoencoders.

#### 6.4.1 Single Frame Feature Extraction

The autoencoder consists of two convolutional layers for spatial encoding followed by a LSTM convolutional layer for temporal encoding. As a general rule of thumb, we use small filters (3x3 and at most 5x5), except for the very first convolutional layer connected to the input, and strides of four and two for the two conv layers (Li et al., 2020). The decoder mirrors the encoder to reconstruct the image (Fig. 6.5). After each convolutional layer, we employ layer normalization, which is equivalent to batch normalization for recurrent networks (Ba et al., 2016). It normalizes the inputs across features and is defined as:

$$\mu_j = \frac{1}{m} \sum_{i=1}^m x_{ij}; \sigma_j^2 = \frac{1}{m} \sum_{i=1}^m (x_{ij} - \mu_j)^2; \hat{x}_{ij} = \frac{x_{ij} - \mu_j}{\sqrt{\sigma_j^2 + \epsilon}} \quad (6.3)$$

where  $x_{ij}$  is batch  $i$ , feature  $j$  of the input  $x$  across  $m$  features.

#### 6.4.2 Frame Aggregation

In chronological order, the training video frames make a temporal sequence. LSTM convolutional layers are used to preserve content and spatial information. To generate more training sequences, we perform data augmentation by concatenating frames with strides 1, 2, and 3. For example, we use a skipping stride of 2 to generate a sequence of every other frame. We use a 10-frame sequence size as a sliding window technique for aggregation of the encodings. The encoder weights learned here are used to then learn 3D decoder weights to output a 3D voxel reconstruction based on audio-visual inputs from audio-augmented ModelNet with impact sound synthesis and Sound20K video.

#### 6.4.3 Modality Fusion and 3D Decoder

After encoding our inputs with LSTM convolutional layers, we flatten to a fully connected layer for each audio and visual subnetwork. These dense layers are fused together prior to multiple Conv3D trans-

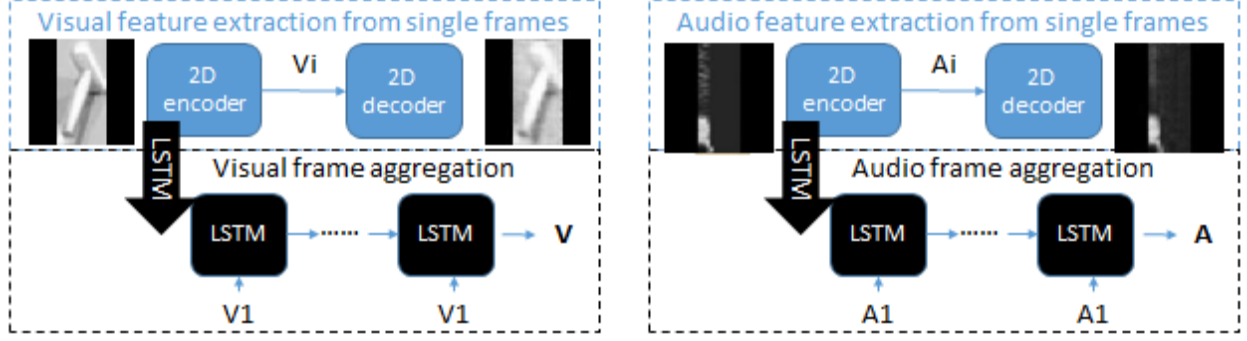


Figure 6.5: Hidden layer representations  $V_i$  and  $A_i$  are trained to spatially encode object geometry and impact sounds, where  $i$  is each video frame. These learned weights are subsequently used during test time to generate 3D shapes from audio-visual inputs. For sequence modeling, LSTM layers are reliable for temporal consistency and establishing dependencies over time. More specifically, we use convolutional LSTM layers rather than fully connected to also preserve spatial information.

pose layers for the 3D decoder. Prior work in multimodal deep learning, such as visual question and answering systems, have merged modalities for classification tasks using addition and MFB (Yu and Tao, 2014). A 3D decoder accepts the fusion of audio-visual LSTM encodings and maps it to a voxel grid with five deconvolutional layers, similar to T-L Network (Girdhar et al., 2016). Unlike T-L’s  $20^3$  voxel grid, we use  $30^3$  voxels for greater resolution and apply a single, audio-based material classification to all voxels. Deconvolution, also known as fractionally-strided or transposed convolution, results in a 3D voxel shape by broadcasting input  $X$  through kernel  $K$  (Zhang et al., 2020).

$$\sum_{i=0}^h \sum_{j=0}^w Y[i : i + h, j : j + w] += (X[i, j] * K) \quad (6.4)$$

## 6.5 Results

In this section, we present our implementation, training, and evaluation metrics along with 3D-MOV reconstructed objects (Fig. 6.8). Please see a comparative analysis of loss and accuracy against baseline methods by dataset and number of views. For each of the datasets ShapeNet and Sound20K, we evaluate the network architecture described in Section 6.4 against audio, visual, and audio-visual methods using binary cross entropy loss and intersection over union (IoU) reconstruction accuracy.

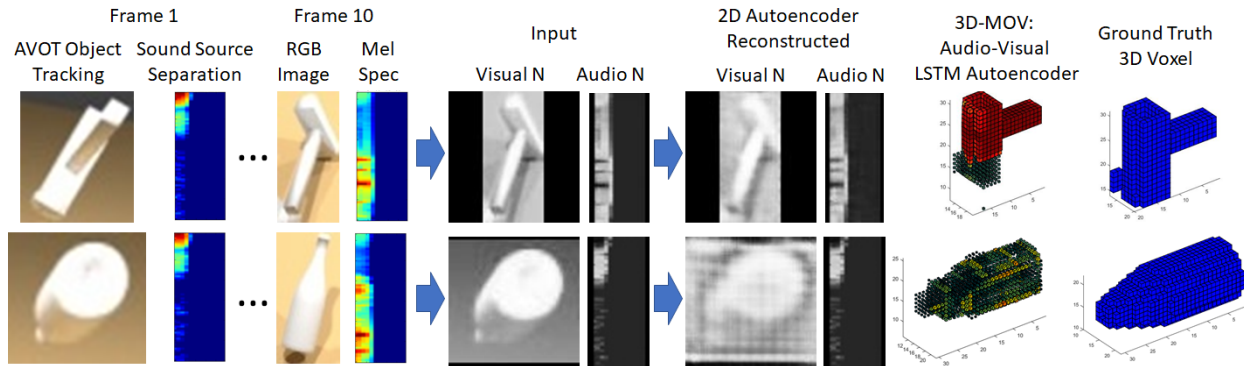


Figure 6.6: Reconstructed objects from using multiple frames and impact sounds. Please see our supplementary materials for a complete review of results for ShapeNet and Sound20K datasets using binary cross entropy loss and reconstruction accuracy comparing audio, visual, and audio-visual methods by number of views. Our method is able to obtain better reconstruction results for concave internal structures and scenes with multiple objects by fusing temporal audio-visual inputs.

### 6.5.1 Implementation

Our framework was implemented using Tensorflow (Abadi et al., 2015) and Keras (Chollet et al., 2015). Training was run on Ubuntu 16.04.6 LTS with a single Titan X GPU. Voxel representations were rendered based on Matlab visualization code from 3D-GAN (Wu et al., 2016). From Sound20K videos, images were grayscale with dimensions  $84 \times 84 \times 1$  and audio spectrograms were  $64 \times 25 \times 1$ , zero padded to equivalent dimensions. Visual data was augmented with resizing, cropping, and skipping strides.

### 6.5.2 Training

Since joint optimization can be difficult to perform, we train our reconstruction autoencoder and fused audio-visual networks separately and then jointly optimize to fine-tune the final network. Mean square error is used for the 2D reconstruction loss to train the encoder to reconstruct input images and audio spectrograms. Binary cross entropy loss is calculated between ground truth and reconstructed 3D voxel grids. During testing, we reconstruct from encoded vector representation of audio-visual inputs to a 3D voxel reconstruction output.

Previous work has used symmetry induced volume refinement to constrain and finalize GAN volumetric outputs (Niu et al., 2018). Other methods have used multiple views to continuously refine the output (Choy et al., 2016). Furthermore, most adversarial generating methods create examples by perturbing

existing data, limiting the solution space. Our approach constrains the space of possible 3D reconstructions for objects in the scene by temporal consistency, aggregation, and fusion of audio and visual inputs.

### 6.5.3 Evaluation metrics

Methods were evaluated against voxel Intersection-over-Union (IoU), also known as the Jaccard index (Jaccard, 1901), between the 3D reconstruction and ground truth voxels as well as cross-entropy loss. This can be represented as area of overlap divided by the area of union. More formally:

$$IoU = \frac{\sum_{i,j,k} [I(p_{(i,j,k)} > t) I(y_{(i,j,k)})]}{\sum_{i,j,k} [I(I(p_{(i,j,k)} > t) + I(y_{(i,j,k)}))]} \quad (6.5)$$

where  $y_{i,j,k} \in 0, 1$  is the ground truth occupancy,  $p_{i,j,k}$  the Bernoulli distribution output at each voxel,  $I(\cdot)$  an indicator function, and  $t$  for threshold. Higher IoU means better reconstruction quality.

Please see Table 6.1 for results by dataset against baseline methods: Figure 6.7 for example ModelNet10 reconstructions and Figure 6.8 for exemplary Sound20K reconstructions, and Figure 6.9 for training loss for audio, visual, and audio-visual.

## 6.6 ML Reproducibility

In this section of our supplementary document, we discuss setup of our experiments and datasets for reproducibility and future research.

### 6.6.1 Experimental Results

Average runtime for 3D-MOV audio-visual training of the Sound20K dataset was about 1.5 minutes per epoch for a sequence size of 10 and strides of 1, 2, and 3. Using 20 epochs, average training time was 30 minutes. For sequence size of 5, 10k training examples took about 4 minutes per. For sequence size of 1, 50k training examples completed in roughly 10 minutes per epoch. Methods were evaluated against voxel Intersection-over-Union (IoU), also known as the Jaccard index (Jaccard, 1901), between the 3D reconstruction and ground truth voxels as well as cross-entropy loss.

Table 6.1: 3D-MOV was evaluated against baselines for loss (mean square error and binary cross entropy) and reconstruction accuracy (intersection over union). A view consists of both an image and audio frame. Decreases in 3D-MOV accuracy as ShapeNet views increase requires further investigation but may suggest impact sounds of different hit points are needed rather than using the same sound across views. \*We use the T-L Network (Girdhar et al., 2016) fused with audio as an overall baseline comparator with 0.67 loss and 18.0% IoU for an instance of the MN10 chair class. \*\* Reported in (Choy et al., 2016)

Dataset Method	Input	ShapeNet (Chang et al., 2015)		Sound20K (Zhang et al., 2017d)
		1 view	5 views	10 views
<b>3D-MOV-A (Ours)</b>	A	21.2%	N/A	37.15%
3D-R2N2 (Choy et al., 2016)	V	56.0%**	63.1%**	N/A
<b>3D-MOV-V (Ours)</b>	V	22.7%	22.5%	65.7%
T-L Network (Girdhar et al., 2016)	AV	18.0%*	N/A	N/A
<b>3D-MOV-AV (Ours)</b>	AV	32.6%	31.0%	69.8%

## 6.6.2 Datasets

We use a server with Ubuntu 16.04.6 LTS and a single Titan X GPU. Training and hold-out test splits were 80% and 20% respectively. With Sound20K sequence size of 10 and strides 1-3, we have 9,800 training and 1,960 test examples of RGB image and audio mel-scaled spectrograms. For ModelNet10, we used voxelized objects since the sound synthesis pipeline requires watertight meshes. Downloadable versions of the datasets used can be found for audio-visual Sound20K synthetic videos (Zhang et al., 2017d), ModelNet10 (Chang et al., 2015), and voxelized ModelNet10 with impact sounds (Auston Sterling and Lin, 2018). Future work is to explore the impact of increases in dataset size on performance. Pre-processing of audio involved converting sound files to mel-scaled spectrograms. Each spectrogram is 3 seconds for a single frame (ShapeNet) and 0.03 seconds per multi-frame (Sound20K). For visual data, segmented frames are grayscaled and resized to a consistent input size of 88 by 88 pixels. While resizing, aspect ratio was maintained and padded to square the image. These dimensions were chosen to account for the size of objects in our Sound20K and ModelNet10 datasets to capture their semantic information.

## 6.7 Conclusions

To the best of our knowledge, this is the first method to use audio and visual inputs from ShapeNet objects and Sound20K video of multiple objects in a scene to generate 3D object reconstructions with material. While multi-view approaches can improve reconstruction accuracy, transparent objects, interior

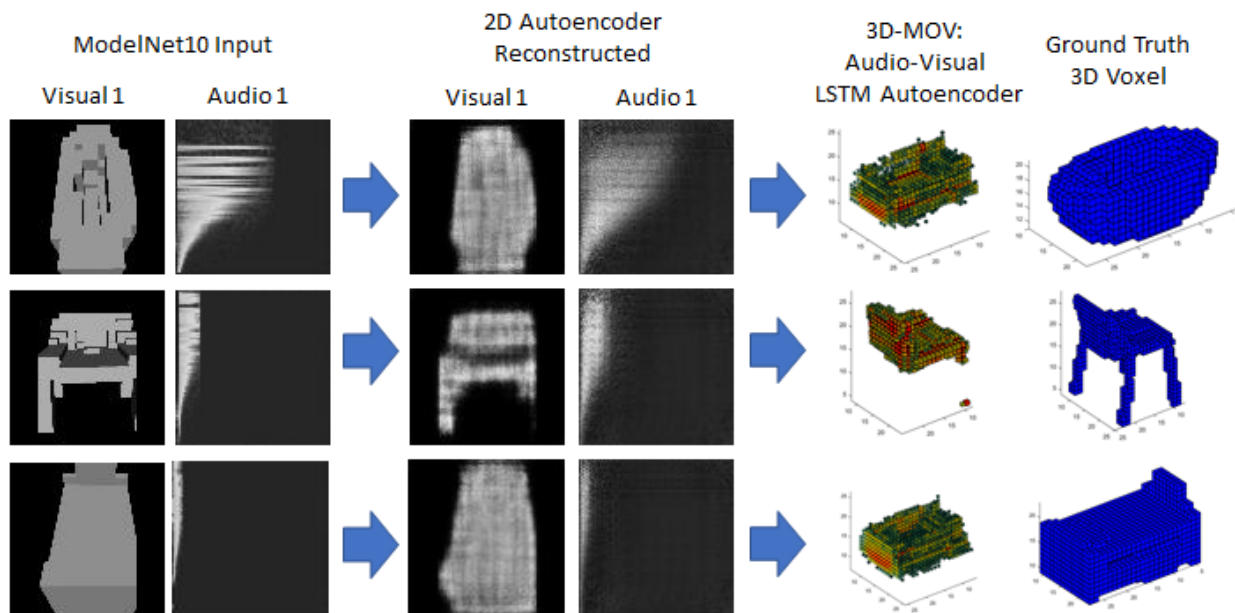


Figure 6.7: 3D-MOV-AV reconstructed image and audio inputs for single view voxelized ModelNet10 classes (top, bathtub; middle, chair; bottom, bed). These results are using a single image and single impact sound, fusing the two modalities with an addition merge layer, training for 60 epochs on a single GPU, and using a voxel threshold of 0.4. 3D-MOV-AV performs the best on ModelNet10 single views, showing audio augmenting visual data.

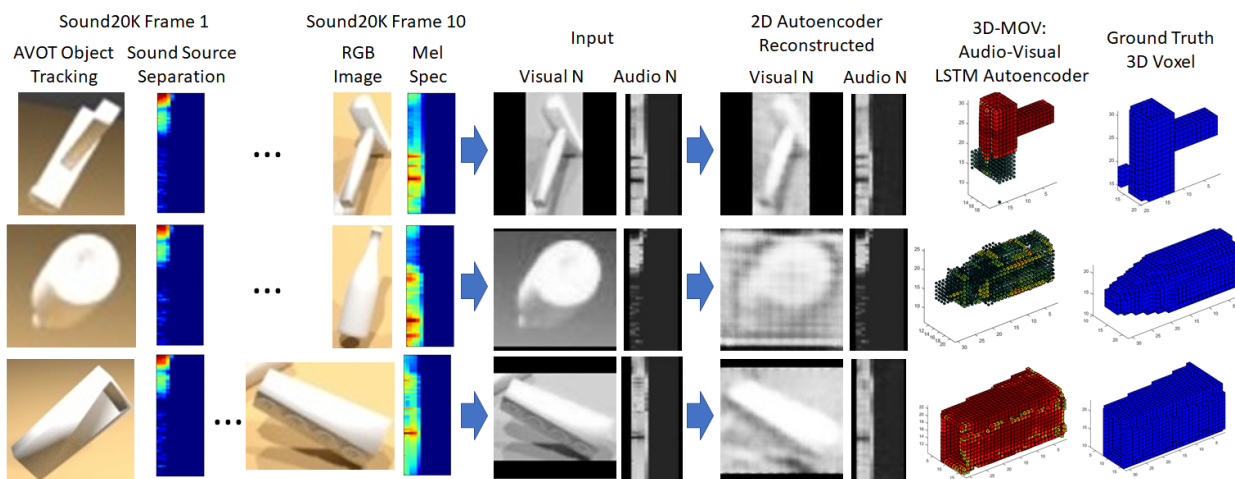


Figure 6.8: Reconstructed objects from using multiple frames and impact sounds. Please see Table 6.1 for results from ShapeNet and Sound20K datasets using binary cross entropy loss and reconstruction accuracy comparing audio, visual, and audio-visual methods by number of views. Audio-visual 3D-MOV-AV performs the best for Sound20K sequence size of 10 views.

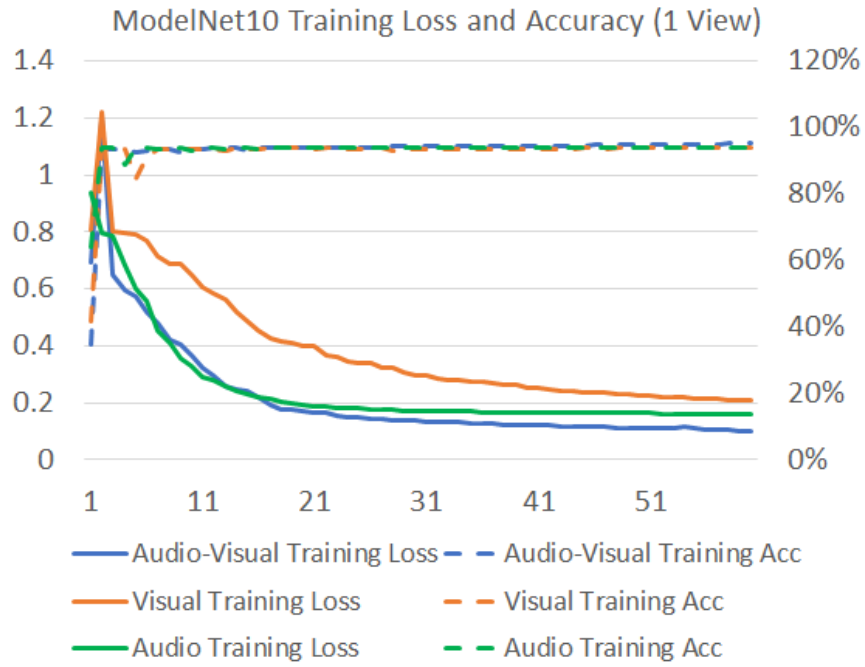


Figure 6.9: Training loss and accuracy for ModelNet10 dataset of 60 epochs. 3D-MOV-AV concludes training with the best performance in terms of binary cross entropy loss and accuracy based on ModelNet10 single views, showing audio augmenting visual data.

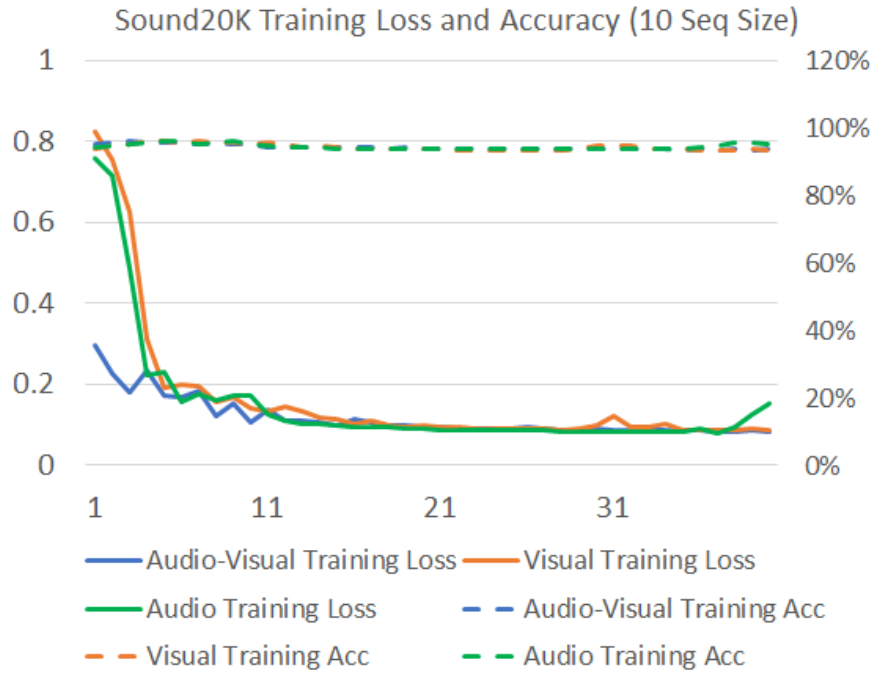


Figure 6.10: Training loss and accuracy for Sound20K dataset of 40 epochs using a sequence size of 10.



concave structures, self-occlusions, and multiple objects remain a challenge. As objects collide, audio provides a complementary sensory input that can enhance the reconstruction model to improve results. In this paper, we demonstrate that augmenting image encodings with corresponding impact sounds refine reconstructions of multimodal LSTM autoencoder neural network outputs.

**Limitations:** our approach is currently implemented and evaluated with fixed-grid shapes. Further experimentation with other resolutions, residual architectures (He et al., 2015a), adaptive grids, and multi-scale reasoning (Denton et al., 2015) are worth exploring. Material classification is predicted based on audio alone, given the textureless image renderings of the datasets used. Also, only a single material is inferred for the entire geometry rather than per voxel classification. Finally, the trade-off between additional views and additional auditory inputs could be further explored.

**Future Work:** evaluation of other real-time object trackers, such as YOLO and Faster R-CNN, can be performed and trained on other existing datasets, such as COCO and SUN RGB-D. Further investigations can also examine how the error introduced by object tracking propagates to reconstruction error. Same applies to errors from sound source separation and being able to accurately associate unmixed sounds with their corresponding visual object tracks. Next, while audio helps classify the material of the reconstructed geometry, we assume a single material classification based on audio alone and apply that to all voxels. Research on classifying material per voxel using both audio and visual data could expand part segmentation research into reconstructing objects with different materials. Rather than being fully deterministic, fusing audio and visual information for generative models to reconstruct geometry and material may also be of interest to the research community. Then, there may be more than one possible 3D reconstruction for a given image or sound. Beyond reconstruction, audio may also enhance image and sound generation, as well as memory and attention models. For instance, image generation using an audio conditioned GAN and sound generation based on image conditioning could be explored, similar to WaveNet (van den Oord et al., 2016a) local and global conditioning techniques. Finally, testing on real data in the wild and larger datasets of annotated audio and visual data allow for continued research in this area.

## CHAPTER 7: SUMMARY AND CONCLUSIONS

This dissertation was motivated by the advances in computer vision and the possibility of realizing the potential benefits of single model audio and multimodal audio-visual learning. Whether by coupling a fluid and structure to form a rigid double body for sound synthesis or fusing audio and visual inputs for object classification, tracking, and reconstruction, audio is readily available for use along with its corresponding images when datasets are generated from video.

Certain conditions also lend themselves more preferably to one modality or a combination of multiple modes. For example, vision-based methods are sufficient for most static objects and scenes. However, reflective and textureless surfaces may be better suited for audio methods since visual data may be ambiguous or changing over time and viewpoint. Finally, audio-visual techniques can use scheduling to use the appropriate inputs given the current state, account for drift error of dynamic objects and scenes, and handle occlusions from cluttered scenes.

### 7.1 Summary of Results

I have presented a fast and practical method for simulating the sound of non-empty objects containing fluids. This work enhanced the sound synthesis equation in the rigid body audio pipeline method and was demonstrated for use in interactive 3D systems, where live sound synthesis is important. The key contribution was to account for the fluid force on an object at the fluid-structure boundary. This was achieved by adding pre-processing steps to identify the mesh nodes of a tetrahedralized object that are in contact with the liquid and to apply an added mass operator to those structural boundary nodes and adjacent solid domain nodes. The added mass is applied to the bounding elements in the mass matrix proportional to the liquid's volume and density, which may vary with temperature and/or type of fluid. The technique generalizes to any impermeable tetrahedral mesh representing the rigid objects and inviscid liquids.

To estimate the weight of a liquid poured into a target container, perform overflow detection, and classify liquid and target container, I introduced a novel audio-based and audio-augmented techniques,

Fluid-structure coupling used added mass operator for sound synthesis
Pouring Sequence Neural Network (PSNN) for weight estimation of liquid
Audio-Visual Object Tracker (AVOT)
Echo-Reconstruction: audio-augmented scene reconstruction on mobile devices
3D-MOV: audio-visual LSTM autoencoder for 3D reconstruction of multiple objects from video

Table 7.1: Summary of contributions

in the form of multimodal convolutional neural networks (CNNs). The audio-based neural network uses the sound from a pouring sequence—a liquid being poured into a target container. Audio inputs consist of converting raw audio into mel-scaled spectrograms. Our audio-augmented network fuses this audio with its corresponding visual data based on video images. Only a microphone and camera are required, which can be found in any modern smartphone or Microsoft Kinect. Our approach improves classification accuracy for different environments, containers, and contents of the robot pouring task. Our Pouring Sound Neural Networks (PSNN) are trained and tested using the Rethink Robotics Baxter Research Robot. To the best of our knowledge, this is the first use of audio-visual neural networks to analyze liquid pouring sequences by classifying their weight, liquid, and receiving container.

Existing state-of-the-art object tracking can run into challenges when objects collide, occlude, come close to one another, or appear similar but are of different materials. By using audio of the impact sounds from object collisions, rolling, etc., I presented an audio-visual object tracking (AVOT) neural network that can reduce tracking error and drift. AVOT is trained end to end and uses audio-visual inputs over all frames. Our audio-based technique may be used in conjunction with other neural networks to augment

visually based object detection and tracking methods. It is evaluated in terms of runtime frames-per-second (FPS) performance and intersection over union (IoU) performance against OpenCV object tracking implementations and a deep learning method. Experiments include using the synthetic Sound-20K audio-visual dataset and demonstrating that AVOT outperforms single-modality deep learning methods, when there is audio from object collisions. A proposed scheduler network to switch between AVOT and other methods based on audio onset maximizes accuracy and performance over all frames in multimodal object tracking.

I proposed "*Echoreconstruction*", an audio-visual method that uses the reflections of sound to aid in geometry and audio reconstruction. This system aids in reconstructing reflective and textureless surfaces such as windows, mirrors, and walls that are often poorly reconstructed and filled with depth discontinuities and holes. The mobile phone prototype emits pulsed audio, while recording video for RGB-based 3D reconstruction and audio-visual classification. Reflected sound and images from the video are input into our audio (EchoCNN-A) and audio-visual (EchoCNN-AV) convolutional neural networks for surface and sound source detection, depth estimation, and material classification. The inferences from these classifications enhance scene 3D reconstructions containing open spaces and reflective surfaces by depth filtering, inpainting, and placement of unmixed sound sources in the scene.

I proposed a multimodal single- and multi-frame neural network for 3D reconstructions using audio-visual inputs. The trained reconstruction LSTM autoencoder 3D-MOV accepts multiple inputs to account for a variety of surface types and views. The neural network produces high-quality 3D reconstructions using voxel representation. Based on Intersection-over-Union (IoU), it is evaluated against other baseline methods using synthetic audio-visual datasets ShapeNet and Sound20K with impact sounds and bounding box annotations. To the best of our knowledge, our single- and multi-frame model is the first audio-visual reconstruction neural network for 3D geometry and material representation.

## 7.2 Limitations and Future Work

Overall, the immediate next research steps to further enhance audio-visual performance and processing is further analysis of tasks that can capture audio in their datasets and benefit from its signal (e.g. inference, tracking, reconstruction), gating or scheduling of when single or multiple modalities are used, more neural network architectures, loss functions, and fusion models, and augmenting with even more modes.

Sound synthesis for fluid-structure coupling has limitations in the form of simplifications to maintain interactive performance in VR applications. First, while the work assumes that liquids are inviscid, remain steady, and are not mixed, it should be extensible to handle mixed fluids. This remains future work to evaluate. Next, the granularity of the solid mesh discretization also influences the results since the modifications to the mass matrix occur at the level of the mesh nodes. Finally, investigation of acoustic transfer, harmonic pressure, and user evaluation on auditory perception would offer additional insight.

Future directions for analyzing liquid pouring sequences include data augmentation to improve classification accuracy and generalization. As the task involves temporal data, sequential layers can be introduced into the neural network model, such as recurrent, LSTM, or GRU layers or HMM filtering and evaluated for performance. This may be especially helpful for audio only PSNN-A classification at the beginning and end of pouring sequences. Using a multiple output neural network rather than separately trained neural networks for poured weight, content, and target container classification may also help as well as using a ratio of volume over the target container volume or a combination. Finally, further research can explore if this approach can be applied to other granular materials.

Future work for audio-visual object tracking may consist of expanding the size of the training set by annotating more objects in the Sound-20K dataset, increasing the number of object classes that we are predicting, evaluating alternative fusion methods, and performing sensitivity analysis on scaling factors and aspect ratios. This object tracking has been used for audio-visual input for 3D reconstruction of tracked objects. Further investigations can examine how the error introduced by object tracking propagates to reconstruction error. Also, while audio helps classify the material of the reconstructed geometry, we assume a single material classification based on audio alone and apply that to all voxels. Research on classifying material per voxel using both audio and visual data could expand part segmentation research into reconstructing objects with different materials.

In addition to object reconstruction, I also presented enhanced scene reconstruction. To further extend this particular area of audio-visual research, a primary focus could be on the reception, and 3D reconstruction simultaneously and in real time instead of having a staged approach. An integrated approach may prove not only to be more efficient but also more effective by using audio feedback as part of the reconstruction code. Another possible avenue of exploration is to investigate the impact of live audio for training and/or testing our neural network variations.

### 7.3 Conclusion

In research and practice, sound is a key contributor to the level of immersion and sense of presence in virtual and imaginative environments. A distraction from any of the senses can cause a ‘break in presence’. A goal in computer graphics is to continue to enhance rendering pipelines with new technology, methods, and data. While vision-based methods cover many use cases, alternate modalities such as audio can augment the level of detail and coverage of tasks in computer vision, graphics, augmented, and virtual reality. Since many sound models are physics-based and training data generated from video, established visual pipelines and datasets can be extended to generate and use sound based on the same physics and video capture used for visual data. Much of the research conducted on visual data is also relevant to sound sources. This presents opportunities for audio-based research to advance quickly based learnings from decades of vision research as well as novel directions for fusing audio, visual, and other data for multimodal learning.

## REFERENCES

- Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. URL <https://www.tensorflow.org/>. Software available from tensorflow.org.
- Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Gregory S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian J. Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Józefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Rajat Monga, Sherry Moore, Derek Gordon Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul A. Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda B. Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *ArXiv*, abs/1603.04467, 2016.
- Jean-Marie Adrien. The missing link: modal synthesis. 1991.
- Ibraheem Alhashim and Peter Wonka. High quality monocular depth estimation via transfer learning. *arXiv e-prints*, abs/1812.11941:arXiv:1812.11941, 2018. URL <https://arxiv.org/abs/1812.11941>.
- N. Anusha and L. Roy. Object tracking from audio and video data using linear prediction method, 2015.
- Relja Arandjelovic and Andrew Zisserman. Look, listen and learn. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 609–617, 2017.
- Relja Arandjelovic and Andrew Zisserman. Objects that sound. *ArXiv*, abs/1712.06651, 2018.
- Relja Arandjelović and Andrew Zisserman. Look, listen and learn, 2017.
- Anurag Arnab, Michael Sapienza, Stuart Golodetz, Julien Valentin, Ondrej Miksik, Shahram Izadi, and Philip Torr. Joint object-material category segmentation from audio-visual cues. In *Proceedings of the British Machine Vision Conference (BMVC)*, 09 2015. doi: 10.5244/C.29.40.
- Matan Atzmon, Haggai Maron, and Yaron Lipman. Point convolutional neural networks by extension operators. *CoRR*, abs/1803.10091, 2018. URL <http://arxiv.org/abs/1803.10091>.
- Sam Lowe Auston Sterling, Justin Wilson and Ming C. Lin. Isnn: Impact sound neural network for audio-visual object classification. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.
- Yusuf Aytar, Carl Vondrick, and Antonio Torralba. Soundnet: Learning sound representations from unlabeled video. *ArXiv*, abs/1610.09001, 2016.
- Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer normalization, 2016.
- Boris Babenko, Ming-Hsuan Yang, and Serge J. Belongie. Visual tracking with online multiple instance learning. *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 983–990, 2009.

- Shaojie Bai, J. Zico Kolter, and Vladlen Koltun. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling, 2018.
- Tadas Baltrusaitis, Chaitanya Ahuja, and Louis-Philippe Morency. Multimodal machine learning: A survey and taxonomy. *CoRR*, abs/1705.09406, 2017. URL <http://arxiv.org/abs/1705.09406>.
- J. Basic. Analytical and numerical computation of added mass in ship vibration analysis, 2012.
- Yuri Bazilevs, Kenji Takizawa, and T. E. Tezduyar. Computational fluid-structure interaction: Methods and applications. 2013.
- Luca Bertinetto, Jack Valmadre, João F. Henriques, Andrea Vedaldi, and Philip H. S. Torr. Fully-convolutional siamese networks for object tracking. *ArXiv*, abs/1606.09549, 2016.
- Michael Bianco, Peter Gerstoft, James Traer, Emma Ozanich, Marie Roch, Sharon Gannot, and Charles Deledalle. Machine learning in acoustics: Theory and applications. *The Journal of the Acoustical Society of America*, 146:3590–3628, 11 2019. doi: 10.1121/1.5133944.
- David S. Bolme, J. Ross Beveridge, Bruce A. Draper, and Yui Man Lui. Visual object tracking using adaptive correlation filters. *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 2544–2550, 2010.
- Davide Boscaini, Jonathan Masci, Emanuele Rodolà, and Michael M. Bronstein. Learning shape correspondence with anisotropic convolutional neural networks. *CoRR*, abs/1605.06437, 2016. URL <http://arxiv.org/abs/1605.06437>.
- Léon Bottou. Large-scale machine learning with stochastic gradient descent. In Yves Lechevallier and Gilbert Saporta, editors, *Proceedings of the 19th International Conference on Computational Statistics (COMPSTAT'2010)*, pages 177–187, Paris, France, 01 2010. Springer. doi: 10.1007/978-3-7908-2604-3\_16. URL <http://leon.bottou.org/papers/bottou-2010>.
- J. P. Boyd. Chebyshev and fourier spectral methods: Second revised edition, 2 revised ed. Dover Publications, December 2001.
- E. Boyer. Continuous auditory feedback for sensorimotor learning, 2015.
- G. Bradski. The opencv library, 2000.
- Carlos Alberto Brebbia and Robert D. Ciskowski. Boundary element methods in acoustics. 1991.
- Christopher Earls Brennen. A review of added mass and fluid inertial forces. 1982.
- Van Brummelen. Added mass effects of compressible and incompressible flows in fluid-structure interaction. *Journal of Applied Mechanics*, 76:021206, 2009.
- Remi Cadene, Hedi Ben-younes, Matthieu Cord, and Nicolas Thome. Murel: Multimodal relational reasoning for visual question answering, 2019.
- Sergi Caelles, Jordi Pont-Tuset, Federico Perazzi, Alberto Montes, Kevis-Kokitsi Maninis, and Luc Van Gool. The 2019 davis challenge on vos: Unsupervised multi-object segmentation. *ArXiv*, abs/1905.00737, 2019.
- MECHANICAL ENGINEERING DEPARTMENT CAL POLY POMONA. Fluid mechanics: Topic 4.3 - hydrostatic force on a curved surface, 2016.



- Rohan Chabra, Julian Straub, Chris Sweeney, Richard A. Newcombe, and Henry Fuchs. Stereodrnnet: Dilated residual stereo net. *CoRR*, abs/1904.02251, 2019. URL <http://arxiv.org/abs/1904.02251>.
- Jeffrey N. Chadwick and Doug L. James. Animating fire with sound. *ACM Trans. Graph.*, 30:84, 2011.
- Angel X. Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. Shapenet: An information-rich 3d model repository, 2015.
- Neal Checka, Kevin Wilson, and Vibhav Rangarajan. Person tracking using audio-video sensor fusion. 2001.
- Kai Chen, Jiaqi Wang, Shuo Yang, Xingcheng Zhang, Yuanjun Xiong, Chen Change Loy, and Dahua Lin. Optimizing video object detection via a scale-time lattice. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7814–7823, 2018.
- François Chollet et al. Keras. <https://keras.io>, 2015.
- François Chollet. keras. <https://github.com/keras-team/keras>, 2015.
- Yong Shean Chong and Yong Haur Tay. Abnormal event detection in videos using spatiotemporal autoencoder, 2017.
- Christopher B Choy, Danfei Xu, JunYoung Gwak, Kevin Chen, and Silvio Savarese. 3d-r2n2: A unified approach for single and multi-view 3d object reconstruction. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2016.
- Samuel Clarke, Travers Rhodes, Christopher G. Atkeson, and Oliver Kroemer. Learning audio feedback for estimating amount and flow of granular material. In *CoRL*, 2018.
- Nikolaus Correll, Kostas E. Bekris, Dmitry Berenson, Oliver Brock, Albert Causo, Kris K. Hauser, Kei Okada, Alberto Rodriguez, Joseph M. Romano, and Peter R. Wurman. Analysis and observations from the first amazon picking challenge. *IEEE Transactions on Automation Science and Engineering*, 15:172–188, 2018.
- Thomas M. Cover and Peter E. Hart. Nearest neighbor pattern classification. *IEEE Trans. Inf. Theory*, 13: 21–27, 1967.
- Michael Cowling and Renate Sitte. Comparison of techniques for environmental sound recognition. *Pattern Recognition Letters*, 24(15):2895 – 2907, 2003. ISSN 0167-8655. doi: [https://doi.org/10.1016/S0167-8655\(03\)00147-8](https://doi.org/10.1016/S0167-8655(03)00147-8). URL <http://www.sciencedirect.com/science/article/pii/S0167865503001478>.
- Marco Crocco, Andrea Trucco, and Alessio Del Bue. Uncalibrated 3d room reconstruction from sound. *CoRR*, abs/1606.06258, 2016. URL <http://arxiv.org/abs/1606.06258>.
- James Cummings and Jeremy Bailenson. How immersive is enough? a meta-analysis of the effect of immersive technology on user presence. *Media Psychology*, 19:1–38, 05 2015. doi: 10.1080/15213269.2015.1015740.
- Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proc. Computer Vision and Pattern Recognition (CVPR), IEEE*, 2017a.

- Angela Dai, Matthias Nießner, Michael Zollhöfer, Shahram Izadi, and Christian Theobalt. Bundlefusion: real-time globally consistent 3d reconstruction using on-the-fly surface re-integration. *ACM Transactions on Graphics*, 36:1, 07 2017b. doi: 10.1145/3072959.3126814.
- Angela Dai, Daniel Ritchie, Martin Bokeloh, Scott Reed, Jürgen Sturm, and Matthias Nießner. Scancomplete: Large-scale scene completion and semantic segmentation for 3d scans. *CoRR*, abs/1712.10215, 2017c. URL <http://arxiv.org/abs/1712.10215>.
- J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.
- Emily L. Denton, Soumith Chintala, Arthur Szlam, and Robert Fergus. Deep generative image models using a laplacian pyramid of adversarial networks. *CoRR*, abs/1506.05751, 2015. URL <http://arxiv.org/abs/1506.05751>.
- Chau Do, Tobias Schubert, and Wolfram Burgard. A probabilistic approach to liquid level detection in cups using an rgb-d camera. pages 2075–2080, 10 2016. doi: 10.1109/IROS.2016.7759326.
- Yoshinori Dobashi, Tsuyoshi Yamamoto, and Tomoyuki Nishita. Real-time rendering of aerodynamic sound using sound textures based on computational fluid dynamics. *ACM Trans. Graph.*, 22:732–740, 2003.
- Jernej Barbic Doug L. James and Dinesh K. Pai. Precomputed acoustic transfer: Output-sensitive, accurate sound generation for geometrically complex vibration sources. In *ACM Transactions on Graphics*, 2006.
- M. D. Egan. *Architectural Acoustics*. McGraw-Hill Custom Publishing, 1988.
- David Eigen and Rob Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. *CoRR*, abs/1411.4734, 2014. URL <http://arxiv.org/abs/1411.4734>.
- Ariel Ephrat, Inbar Mosseri, Oran Lang, Tali Dekel, Kevin Wilson, Avinatan Hassidim, William T. Freeman, and Michael Rubinstein. Looking to listen at the cocktail party: A speaker-independent audio-visual model for speech separation. *ACM Trans. Graph.*, 37(4), July 2018a. ISSN 0730-0301. doi: 10.1145/3197517.3201357. URL <https://doi.org/10.1145/3197517.3201357>.
- Ariel Ephrat, Inbar Mosseri, Oran Lang, Tali Dekel, Kevin Wilson, Avinatan Hassidim, William T. Freeman, and Michael Rubinstein. Looking to listen at the cocktail party: A speaker-independent audio-visual model for speech separation. *CoRR*, abs/1804.03619, 2018b. URL <http://arxiv.org/abs/1804.03619>.
- M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. <http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html>.
- Haoqiang Fan, Hao Su, and Leonidas J Guibas. A point set generation network for 3d object reconstruction from a single image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 605–613, 2017.
- Akira Fukui, Dong Huk Park, Daylen Yang, Anna Rohrbach, Trevor Darrell, and Marcus Rohrbach. Multimodal compact bilinear pooling for visual question answering and visual grounding. *CoRR*, abs/1606.01847, 2016. URL <http://arxiv.org/abs/1606.01847>.

- Thomas Funkhouser, Nicolas Tsingos, and Jean-Marc Jot. Survey of methods for modeling sound propagation in interactive virtual environment systems, 2003.
- Yang Gao, Oscar Beijbom, Ning Zhang, and Trevor Darrell. Compact bilinear pooling, 2015.
- Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3354–3361, 2012.
- Jort Gemmeke, Daniel Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R. Moore, Manoj Plakal, and Marvin Ritter. Audio set: An ontology and human-labeled dataset for audio events. In *Proc. IEEE ICASSP 2017*, pages 776–780, New Orleans, LA, 03 2017. doi: 10.1109/ICASSP.2017.7952261.
- Rohit Girdhar, David F. Fouhey, Mikel Rodriguez, and Abhinav Gupta. Learning a predictable and generative vector representation for objects. *CoRR*, abs/1603.08637, 2016. URL <http://arxiv.org/abs/1603.08637>.
- Daniel Godoy, Bashima Islam, Stephen Xia, Md Tamzeed Islam, Rishikanth Chandrasekaran, Yen-Chun Chen, Shahriar Nirjon, Peter Kinget, and Xiaofan Jiang. Paws: A wearable acoustic system for pedestrian safety. pages 237–248, 04 2018. doi: 10.1109/IoTDL.2018.00031.
- Stuart Golodetz\*, Michael Sapienza\*, Julien P C Valentin, Vibhav Vineet, Ming-Ming Cheng, Anurag Arnab, Victor A Prisacariu, Olaf Kähler, Carl Yuheng Ren, David W Murray, Shahram Izadi, and Philip H S Torr. SemanticPaint: A Framework for the Interactive Segmentation of 3D Scenes. Technical Report TVG-2015-1, Department of Engineering Science, University of Oxford, October 2015. Released as arXiv e-print 1510.03727.
- Stefan Gottschalk and Ming C. Lin. Collision detection between geometric models: A survey. 1998.
- Stefan Gottschalk, Ming C. Lin, and Dinesh Manocha. Obbtree: a hierarchical structure for rapid interference detection. In *SIGGRAPH '96*, 1996.
- Shane Griffith, Vladimir Sukhoy, Todd Wegter, and Alexander Stoytchev. Object categorization in the sink : Learning behavior – grounded object categories with water. 2012.
- JunYoung Gwak, Christopher B. Choy, Animesh Garg, Manmohan Chandraker, and Silvio Savarese. Weakly supervised generative adversarial networks for 3d reconstruction. *CoRR*, abs/1705.10904, 2017. URL <http://arxiv.org/abs/1705.10904>.
- Tor Halmrast. A very simple way to simulate the timbre of flutter echoes in spatial audio. 2019.
- Zhizhong Han, Xiyang Wang, Yu-Shen Liu, and Matthias Zwicker. Multi-angle point cloud-vae: Un-supervised feature learning for 3d point clouds from multiple angles by joint self-reconstruction and half-to-half prediction, 2019.
- Darryl Hannan, Akshay Jain, and Mohit Bansal. Mnymodalqa: Modality disambiguation and qa over diverse inputs, 2020.
- Mahmudul Hasan, Jonghyun Choi, Jan Neumann, Amit K. Roy-Chowdhury, and Larry S. Davis. Learning temporal regularity in video sequences, 2016.
- Anfeng He, Chong Luo, Xinmei Tian, and Wenjun Zeng. Towards a better match in siamese network based visual object tracker. *ArXiv*, abs/1809.01368, 2018.

- Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015a. URL <http://arxiv.org/abs/1512.03385>.
- Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1026–1034, 2015b.
- Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.
- Kaiming He, Ross B. Girshick, and Piotr Dollár. Rethinking imagenet pre-training. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4917–4926, 2019.
- Peter Hedman, Suhub Alsisan, Richard Szeliski, and Johannes Kopf. Casual 3d photography. *SIGGRAPH ASIA*, 2017.
- João F. Henriques, Rui Caseiro, Pedro Martins, and Jorge Batista. High-speed tracking with kernelized correlation filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37:583–596, 2015.
- Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9:1735–1780, 1997.
- D. Holz. Object detection and tracking with audio and optical signals.
- Shih hong Tsai. Customizing an adversarial example generator with class-conditional gans. *CVPR*, 2018.
- Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *ArXiv*, abs/1704.04861, 2017.
- Ruizhen Hu, Zihao Yan, Jingwen Zhang, Oliver van Kaick, Ariel Shamir, and Hui Huang. Predictive and generative neural networks for object functionality. *ACM Transactions on Graphics*, 37:1–13, 07 2018. doi: 10.1145/3197517.3201287.
- Gao Huang, Zhuang Liu, and Kilian Q. Weinberger. Densely connected convolutional networks. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2261–2269, 2017.
- Jie Huang, Tadawute Supaongprapa, Ikutaka Terakura, Fuming Wang, Noboru Ohnishi, and Noboru Sugie. A model-based sound localization system and its application to robot navigation. *Robotics Auton. Syst.*, 27:199–209, 1999.
- Xun Huang, Ming-Yu Liu, Serge Belongie, and Jan Kautz. Multimodal unsupervised image-to-image translation. *ECCV*, 2018.
- M. Huzaifah. Comparison of time-frequency representations for environmental sound classification using convolutional neural networks, 2017a.
- Muhammad Huzaifah. Comparison of time-frequency representations for environmental sound classification using convolutional neural networks. *CoRR*, abs/1706.07156, 2017b. URL <http://arxiv.org/abs/1706.07156>.

- Ilija Ilievski and Jiashi Feng. Multimodal learning and reasoning for visual question answering. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 551–562. Curran Associates, Inc., 2017. URL <http://papers.nips.cc/paper/6658-multimodal-learning-and-reasoning-for-visual-question-answering.pdf>.
- Shahram Izadi, David Kim, Otmar Hilliges, David Molyneaux, Richard Newcombe, Pushmeet Kohli, Jamie Shotton, Steve Hodges, Dustin Freeman, Andrew Davison, and Andrew Fitzgibbon. Kinectfusion: Real-time 3d reconstruction and interaction using a moving depth camera. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology*, UIST '11, page 559–568, New York, NY, USA, 2011. Association for Computing Machinery. ISBN 9781450307161. doi: 10.1145/2047196.2047270. URL <https://doi.org/10.1145/2047196.2047270>.
- P. Jaccard. Distribution de la flore alpine dans le bassin des drouces et dans quelques regions voisines, 1901.
- Lakhmi C. Jain and Larry R. Medsker. Recurrent neural networks: Design and applications. 1999.
- Doug L. James. Physically based sound for computer animation and virtual environments. In *SIGGRAPH '16*, 2016.
- Doug L. James, Jernej Barbic, and Dinesh K. Pai. Precomputed acoustic transfer: output-sensitive, accurate sound generation for geometrically complex vibration sources. In *SIGGRAPH 2006*, 2006a.
- Doug L. James, Jernej Barbič, and Dinesh K. Pai. Precomputed acoustic transfer: Output-sensitive, accurate sound generation for geometrically complex vibration sources. In *ACM SIGGRAPH 2006 Papers*, SIGGRAPH '06, pages 987–995, New York, NY, USA, 2006b. ACM. ISBN 1-59593-364-6. doi: 10.1145/1179352.1141983. URL <http://doi.acm.org/10.1145/1179352.1141983>.
- A. Janoch, S. Karayev, Y. Jia, J. T. Barron, M. Fritz, K. Saenko, and T. Darrell. A category-level 3-d object dataset: Putting the kinect to work. *ICCV*, 2011.
- Martin Kada and Laurence Mckinley. 3d building reconstruction from lidar based on a cell decomposition approach. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 38, 09 2009.
- Hansung Kim, Luca Remaggi, Philip J. B. Jackson, Filippo Maria Fazi, and Adrian Hilton. 3d room geometry reconstruction using audio-visual sensors. pages 621–629, 10 2017. doi: 10.1109/3DV.2017.00076.
- Hyoungun Kim, Hao Tan, and Mohit Bansal. Modality-balanced models for visual dialogue, 2020.
- Jin-Hwa Kim, Sang-Woo Lee, Dong-Hyun Kwak, Min-Oh Heo, Jeonghee Kim, Jung-Woo Ha, and Byoung-Tak Zhang. Multimodal residual learning for visual qa, 2016.
- Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2014. URL <http://arxiv.org/abs/1412.6980>. cite arxiv:1412.6980Comment: Published as a conference paper at the 3rd International Conference for Learning Representations, San Diego, 2015.
- Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2015.

- Jane Klemen and Christopher D. Chambers. Current perspectives and methods in studying neural mechanisms of multisensory interactions. *Neuroscience & Biobehavioral Reviews*, 36(1):111 – 133, 2012. ISSN 0149-7634. doi: <https://doi.org/10.1016/j.neubiorev.2011.04.015>. URL <http://www.sciencedirect.com/science/article/pii/S0149763411000819>.
- Takashi Konno, Kenji Nishida, Katsutoshi Itoyama, and Kazuhiro Nakadai. Audio-visual 3d reconstruction framework for dynamic scenes. In *Proceedings of the 2020 IEEE/SICE International Symposium on System Integration (SII 2020)*, pages 802–807, Hawaii Convention Center, Honolulu, Hawaii, USA, Jan. 2020. IEEE. URL <https://sice-si.org/conf/SII2020/>.
- Johannes Kopf, Fabian Langguth, Daniel Scharstein, Richard Szeliski, and Michael Goesele. Image-based rendering in the gradient domain. *ACM Transactions on Graphics (TOG)*, 32:199:1–199:9, 11 2013. doi: 10.1145/2508363.2508369.
- Alejandro Koretzky, Karthiek Reddy Bokka, and Naveen Sasalu Rajashekharappa. Real-time adaptive audio source separation, 2017. URL <https://patents.google.com/patent/US20170061978>.
- Stephen Kosslyn. Mental images and the brain. *Cognitive neuropsychology*, 22:333–47, 05 2005. doi: 10.1080/02643290442000130.
- Matej Kristan, Juan E. Sala Matas, Alevs. Leonardis, Tomás Vojír, Roman P. Pflugfelder, Gustavo Fernández, Georg Nebehay, Fatih Murat Porikli, and Luka Cehovin. A novel performance evaluation methodology for single-target trackers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38:2137–2155, 2016.
- Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012a.
- Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012b. URL <http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf>.
- John D. Lafferty, Andrew McCallum, and Fernando Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML*, 2001.
- Kevin Lai, Liefeng Bo, Xiaofeng Ren, and Dieter Fox. A large-scale hierarchical multi-view rgb-d object dataset. pages 1817–1824, 05 2011. doi: 10.1109/ICRA.2011.5980382.
- R. B. Lawson. Pitch perception. technical note 7-65., 1965.
- Colin Lea, Michael D. Flynn, René Vidal, Austin Reiter, and Gregory D. Hager. Temporal convolutional networks for action segmentation and detection. *CoRR*, abs/1611.05267, 2016. URL <http://arxiv.org/abs/1611.05267>.
- Colin Lea, Michael D. Flynn, René Vidal, Austin Reiter, and Gregory D. Hager. Temporal convolutional networks for action segmentation and detection. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1003–1012, 2017.
- Yann LeCun and Yoshua Bengio. Convolutional networks for images, speech, and time series. 1998.

- Daniel D. Lee and H. Sebastian Seung. Algorithms for non-negative matrix factorization. In *Proceedings of the 13th International Conference on Neural Information Processing Systems, NIPS'00*, pages 535–541, Cambridge, MA, USA, 2000. MIT Press. URL <http://dl.acm.org/citation.cfm?id=3008751.3008829>.
- Michelle A. Lee, Yuke Zhu, Peter Zachares, Matthew Tan, Krishnan Srinivasan, Silvio Savarese, Li Fei-Fei, Animesh Garg, and Jeannette Bohg. Making sense of vision and touch: Learning multimodal representations for contact-rich tasks, 2019.
- Bo Li, Junjie Yan, Wei Wu, Zheng Zhu, and Xiaolin Hu. High performance visual tracking with siamese region proposal network. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8971–8980, 2018.
- Fei-Fei Li, Ranjay Krishna, and Danfei Xu. Convolutional neural networks for visual recognition, 2020. URL <https://cs231n.github.io/convolutional-networks/>.
- Ji Lin, Chuang Gan, and Song Han. Tsm: Temporal shift module for efficient video understanding. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 7082–7092, 2019.
- Tsung-Yi Lin, Michael Maire, Serge J. Belongie, Lubomir D. Bourdev, Ross B. Girshick, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. *CoRR*, abs/1405.0312, 2014a. URL <http://arxiv.org/abs/1405.0312>.
- Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. *ArXiv*, abs/1405.0312, 2014b.
- David B. Lindell, Gordon Wetzstein, and Vladlen Koltun. Acoustic non-line-of-sight imaging. pages 6773–6782, 06 2019. doi: 10.1109/CVPR.2019.00694.
- Mason Liu, Menglong Zhu, Marie White, Yinxiao Li, and Dmitry Kalenichenko. Looking fast and slow: Memory-guided mobile video object detection. *ArXiv*, abs/1903.10172, 2019.
- Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott E. Reed, Cheng-Yang Fu, and Alexander C. Berg. SSD: single shot multibox detector. *CoRR*, abs/1512.02325, 2015. URL <http://arxiv.org/abs/1512.02325>.
- Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott E. Reed, Cheng-Yang Fu, and Alexander C. Berg. Ssd: Single shot multibox detector. In *ECCV*, 2016.
- Marshall Long. *Architectural Acoustics*. Academic Press, 2nd. edition, 2014.
- Alan Lukezic, Tomás Vojír, Luka Cehovin, Jiri Matas, and Matej Kristan. Discriminative correlation filter with channel and spatial reliability. *CoRR*, abs/1611.08461, 2016. URL <http://arxiv.org/abs/1611.08461>.
- Alan Lukezic, Tomás Vojír, Luka Cehovin Zajc, Jiri Matas, and Matej Kristan. Discriminative correlation filter with channel and spatial reliability. In *CVPR*, 2017.
- Zhaoliang Lun, Matheus Gadelha, Evangelos Kalogerakis, Subhransu Maji, and Rui Wang. 3d shape reconstruction from sketches via multi-view convolutional networks. In *3D Vision (3DV), 2017 International Conference on*, pages 67–77. IEEE, 2017.
- Hao Luo, Wenxuan Xie, Xinggang Wang, and Wenjun Zeng. Detect or track: Towards cost-effective video object detection/tracking. In *AAAI*, 2019.

- Ilya Lysenkov, Victor Eruhimov, and Gary R. Bradski. Recognition and pose estimation of rigid transparent objects with a kinect sensor. In *Robotics: Science and Systems*, 2012.
- Wenguang Mao, Mei Wang, and Lili Qiu. Aim: Acoustic imaging on a mobile. In *Proceedings of the 16th Annual International Conference on Mobile Systems, Applications, and Services, MobiSys '18*, page 468–481, New York, NY, USA, 2018. Association for Computing Machinery. ISBN 9781450357203. doi: 10.1145/3210240.3210325. URL <https://doi.org/10.1145/3210240.3210325>.
- Xudong Mao, Qing Li, and Haoran Xie. Aligned: Learning to align cross-domain images with conditional generative adversarial networks. *CVPR*, 2017.
- Eric Martinson and Alan Schultz. Discovery of sound sources by an autonomous mobile robot. *Auton. Robots*, 27:221–237, 10 2009. doi: 10.1007/s10514-009-9123-1.
- Daniel Maturana and Sebastian Scherer. Voxnet: A 3d convolutional neural network for real-time object recognition. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, page 922 – 928, September 2015.
- Matthias Mauch and Simon Dixon. Pyin: A fundamental frequency estimator using probabilistic threshold distributions. pages 659–663, 05 2014. ISBN 978-1-4799-2893-4. doi: 10.1109/ICASSP.2014.6853678.
- Ravish Mehra, Atul Rungta, Abhinav Golas, Ming Lin, and Dinesh Manocha. Wave: Interactive wave-based sound propagation for virtual environments. *Visualization and Computer Graphics, IEEE Transactions on*, 21:434–442, 04 2015a. doi: 10.1109/TVCG.2015.2391858.
- Ravish Mehra, Atul Rungta, Abhinav Golas, Ming C. Lin, and Dinesh Manocha. Wave: Interactive wave-based sound propagation for virtual environments. *IEEE Transactions on Visualization and Computer Graphics*, 21:434–442, 2015b.
- Metashape. Agisoft metashape standard, 2020. URL <https://www.agisoft.com/downloads/installer/>.
- Büchler Michael, Allegro Silvia, Stefan Launer, and Norbert Dillier. Sound classification in hearing aids inspired by auditory scene analysis. *EURASIP Journal on Advances in Signal Processing*, 18, 01 2005. doi: 10.1155/ASP.2005.2991.
- Anton Milan, Laura Leal-Taixé, Ian D. Reid, Stefan Roth, and Konrad Schindler. Mot16: A benchmark for multi-object tracking. *ArXiv*, abs/1603.00831, 2016.
- Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *CoRR*, abs/1411.1784, 2014. URL <http://arxiv.org/abs/1411.1784>.
- Meinard Miller. *Fundamentals of Music Processing: Audio, Analysis, Algorithms, Applications*. Springer Publishing Company, Incorporated, 1st edition, 2015. ISBN 3319219448.
- William Moss, Hengchin Yeh, Jeong-Mo Hong, Ming C. Lin, and Dinesh Manocha. Sounding liquids: Automatic sound synthesis from fluid simulation. *ACM Trans. Graph.*, 29:21:1–21:13, 2010.
- Matthias Müller, Julie Dorsey, Leonard McMillan, Robert Jagnow, and Barbara Cutler. Stable real-time deformations. In *SCA '02*, 2002.
- Matthias Müller, Simon Schirm, Matthias Teschner, Bruno Heidelberger, and Markus H. Gross. Interaction of fluids with deformable solids. *Comp. Anim. Virt. Worlds*, 15:159–171, 2004.



- Arun Asokan Nair, Austin Reiter, Changxi Zheng, and Shree Nayar. Audiovisual zooming: What you see is what you hear. In *Proceedings of the 27th ACM International Conference on Multimedia*, MM '19, pages 1107–1118, New York, NY, USA, 2019. ACM. ISBN 978-1-4503-6889-6. doi: 10.1145/3343031.3351010. URL <http://doi.acm.org/10.1145/3343031.3351010>.
- Alexander Neubeck and Luc Van Gool. Efficient non-maximum suppression. *18th International Conference on Pattern Recognition (ICPR'06)*, 3:850–855, 2006.
- Richard Newcombe, Andrew Davison, Shahram Izadi, Pushmeet Kohli, Otmar Hilliges, Jamie Shotton, Steve Hodges, David Kim, and Andrew Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. pages 127–136, 10 2011. doi: 10.1109/ISMAR.2011.6162880.
- Richard Newcombe, Dieter Fox, and Steven Seitz. Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time. pages 343–352, 06 2015. doi: 10.1109/CVPR.2015.7298631.
- J. N. Newman. Marine hydrodynamics, 1977.
- Jiquan Ngiam, Aditya Khosla, Mingyu Kim, Juhan Nam, Honglak Lee, and Andrew Y. Ng. Multimodal deep learning. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, ICML'11, page 689–696, Madison, WI, USA, 2011a. Omnipress. ISBN 9781450306195.
- Jiquan Ngiam, Aditya Khosla, Mingyu Kim, Juhan Nam, Honglak Lee, and Andrew Y. Ng. Multimodal deep learning. In *ICML*, 2011b.
- Mikael Nilsson, J.s Bartunek, Jörgen Nordberg, and I. Claesson. Human whistle detection and frequency estimation. *Image and Signal Processing, Congress on*, 5:737–741, 05 2008. doi: 10.1109/CISP.2008.415.
- Chengjie Niu, Jun Li, and Kai Xu. Im2struct: Recovering 3d shape structure from a single RGB image. *CoRR*, abs/1804.05469, 2018. URL <http://arxiv.org/abs/1804.05469>.
- James O'Brien, Chen Shen, and Christine Gatchalian. Synthesizing sounds from rigid-body simulations. 06 2002. doi: 10.1145/545261.545290.
- Augustus Odena, Christopher Olah, and Jonathon Shlens. Conditional image synthesis with auxiliary classifier gans. *CVPR*, 2017.
- Edwin Olson. Apriltag: A robust and flexible visual fiducial system. pages 3400 – 3407, 06 2011. doi: 10.1109/ICRA.2011.5979561.
- Andrew Owens and Alexei A. Efros. Audio-visual scene analysis with self-supervised multisensory features. *CoRR*, abs/1804.03641, 2018. URL <http://arxiv.org/abs/1804.03641>.
- M. C. Ozkul, A. Saranli, and Y. Yazicioglu. Acoustic surface perception for improved mobility of legged robots, 2012.
- Zherong Pan, Chonhyon Park, and Dinesh Manocha. Robot motion planning for pouring liquids. In *ICAPS*, 2016.
- E. Park, X. Han, T. L. Berg, and A. C. Berg. Combining multiple sources of knowledge in deep cnns for action recognition. In *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1–8, 2016.
- Eunbyung Park and Alexander C. Berg. Meta-tracker: Fast and robust online adaptation for visual object trackers. In *ECCV*, 2018.

- Ramakanth Pasunuru and Mohit Bansal. Multi-task video captioning with video and entailment generation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1273–1283, Vancouver, Canada, July 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1117. URL <https://www.aclweb.org/anthology/P17-1117>.
- Federico Perazzi, Jordi Pont-Tuset, Brian McWilliams, Luc Van Gool, Markus H. Gross, and Alexander Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 724–732, 2016.
- Luis Perez and Jason Wang. The effectiveness of data augmentation in image classification using deep learning. *ArXiv*, abs/1712.04621, 2017.
- Karol J. Piczak. Esc: Dataset for environmental sound classification. In *Proceedings of the 23rd ACM International Conference on Multimedia*, MM ’15, pages 1015–1018, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3459-4. doi: 10.1145/2733373.2806390. URL <http://doi.acm.org/10.1145/2733373.2806390>.
- Victor Adrian Prisacariu, Olaf Kähler, David W. Murray, and Ian D. Reid. Real-time 3d tracking and reconstruction on mobile phones. *IEEE Transactions on Visualization and Computer Graphics*, 21: 557–570, 2015.
- Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. *arXiv preprint arXiv:1612.00593*, 2016a.
- Charles R Qi, Hao Su, Matthias Nießner, Angela Dai, Mengyuan Yan, and Leonidas J Guibas. Volumetric and multi-view cnns for object classification on 3d data. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5648–5656, 2016b.
- Charles R. Qi, Li Yi, Hao Su, and Leonidas J. Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space, 2017a.
- Charles Ruizhongtai Qi, Wei Liu, Chenxia Wu, Hao Su, and Leonidas J. Guibas. Frustum pointnets for 3d object detection from RGB-D data. *CoRR*, abs/1711.08488, 2017b. URL <http://arxiv.org/abs/1711.08488>.
- Xinyuan Qian, Alessio Brutti, Oswald Lanz, Maurizio Omologo, and A. Cavallaro. Multi-speaker tracking from an audio–visual sensing device. *IEEE Transactions on Multimedia*, 21:2576–2588, 2019.
- Lawrence R. Rabiner. A tutorial on hidden markov models and selected applications. pages 267–296. Morgan Kaufmann Publishers Inc. San Francisco, CA, USA, 1989.
- Nikunj Raghuvanshi and Ming Lin. Interactive sound synthesis for large scale environments. volume 2006, 01 2006. doi: 10.1145/1111411.1111429.
- Nikunj Raghuvanshi, Rahul Narain, and Ming C. Lin. Efficient and accurate sound propagation using adaptive rectangular decomposition. *IEEE Transactions on Visualization and Computer Graphics*, 15: 789–801, 2009.
- Caleb Rascón and Ivan Vladimir Meza Ruiz. Localization of sound sources in robotics: A review. *Robotics Auton. Syst.*, 96:184–210, 2017.
- J.W.S. Rayleigh. The theory of sound, volume two, 1945.

- Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection, 2015a.
- Joseph Redmon, Santosh Kumar Divvala, Ross B. Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. *CoRR*, abs/1506.02640, 2015b. URL <http://arxiv.org/abs/1506.02640>.
- Joseph Redmon, Santosh Kumar Divvala, Ross B. Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 779–788, 2016.
- Scott Reed, Zeynep Akata, Santosh Mohan, Samuel Tenka, Bernt Schiele, and Honglak Lee. Learning what and where to draw. *NIPS*, 2016.
- Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39, 06 2015a. doi: 10.1109/TPAMI.2016.2577031.
- Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster R-CNN: towards real-time object detection with region proposal networks. *CoRR*, abs/1506.01497, 2015b. URL <http://arxiv.org/abs/1506.01497>.
- Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39:1137–1149, 2015c.
- Zhimin Ren, Ravish Mehra, Jason Coposky, and Ming Lin. Tabletop ensemble: touch-enabled virtual percussion instruments. *Proceedings - I3D 2012: ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, 03 2012. doi: 10.1145/2159616.2159618.
- Zhimin Ren, Hengchin Yeh, and Ming Lin. Example-guided physically based modal sound synthesis. *ACM Transactions on Graphics (TOG)*, 32:1, 01 2013a. doi: 10.1145/2421636.2421637.
- Zhimin Ren, Hengchin Yeh, and Ming C. Lin. Example-guided physically based modal sound synthesis. *ACM Trans. Graph.*, 32(1):1:1–1:16, February 2013b. ISSN 0730-0301. doi: 10.1145/2421636.2421637. URL <http://doi.acm.org/10.1145/2421636.2421637>.
- Gernot Riegler, Ali Osman Ulusoy, and Andreas Geiger. Octnet: Learning deep 3d representations at high resolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.
- Davide Rocchesso, Roberto Bresin, and Mikael Fernström. Sounding objects. *Multimedia, IEEE*, 10:42–52, 05 2003. doi: 10.1109/MMUL.2003.1195160.
- Atul Rungta, Carl Schissler, Ravish Mehra, Chris Malloy, Ming Lin, and Dinesh Manocha. Syncopation: Interactive synthesis-coupled sound propagation. *IEEE transactions on visualization and computer graphics*, 22, 01 2016a. doi: 10.1109/TVCG.2016.2518421.
- Atul Rungta, Carl Schissler, Ravish Mehra, Chris Malloy, Ming Lin, and Dinesh Manocha. Syncopation: Interactive synthesis-coupled sound propagation. *IEEE Transactions on Visualization and Computer Graphics*, 22(4):1346–1355, April 2016b. ISSN 1077-2626. doi: 10.1109/TVCG.2016.2518421. URL <https://doi.org/10.1109/TVCG.2016.2518421>.

- Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael S. Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115:211–252, 2015.
- Shinichi Sakamoto, Ayumi Ushiyama, and Hiroshi Nagatomo. Numerical analysis of sound propagation in rooms using the finite difference time domain method. *The Journal of the Acoustical Society of America*, 120:3008, 11 2006. doi: 10.1121/1.4787029.
- Justin Salamon, Christopher Jacoby, and Juan Pablo Bello. A dataset and taxonomy for urban sound research. In *Proceedings of the 22Nd ACM International Conference on Multimedia*, MM ’14, pages 1041–1044, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-3063-3. doi: 10.1145/2647868.2655045. URL <http://doi.acm.org/10.1145/2647868.2655045>.
- M. V. Sanchez-Vives and M. Slater. From presence to consciousness through virtual reality. *Nature Reviews Neuroscience*, 6:332–339, 2005.
- Laurent Scallie, Alejandro Koretzky, Karthiek Reddy Bokka, Naveen Sasalu Rajashekharappa, and Luis Daniel Bernal. Virtual music experiences, 2017. URL <https://patents.google.com/patent/US10325580B2/en?q=US10325580B2>.
- Connor Schenck and Dieter Fox. Visual closed-loop control for pouring liquids. *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2629–2636, 2017.
- Carl Schissler and Dinesh Manocha. Gsound: Interactive sound propagation for games. *Proceedings of the AES International Conference*, 02 2011.
- Carl Schissler, Christian Loftin, and Dinesh Manocha. Acoustic classification and optimization for multi-modal rendering of real-world scenes. *IEEE Transactions on Visualization and Computer Graphics*, 24: 1246–1259, 2018.
- Dirk Schulz, Wolfram Burgard, Dieter Fox, and Armin B. Cremers. Tracking multiple moving targets with a mobile robot using particle filters and statistical data association. *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No.01CH37164)*, 2:1665–1670 vol.2, 2001.
- Steven Seitz, Brian Curless, James Diebel, Daniel Scharstein, and Richard Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. volume 1, pages 519–528, 01 2006. doi: 10.1109/CVPR.2006.19.
- A. Sek and B. Moore. Frequency discrimination as a function of frequency, measured in several ways, 2016.
- A. Shabana. Vibration of discrete and continuous systems, 1997.
- YiChang Shih, Dilip Krishnan, Fredo Durand, and William Freeman. Reflection removal using ghosting cues. pages 3193–3201, 06 2015. doi: 10.1109/CVPR.2015.7298939.
- N. Silberman, D. Holey, P. Kohli, and R. Fergus. Indoor segmentation and support inference from rgb-d images. *ECCV*, 2012a.
- Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgb-d images. In *ECCV*, pages 746–760, 10 2012b. doi: 10.1007/978-3-642-33715-4\_54.
- Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2015.

- Arjun Singh, James Sha, Karthik Narayan, Tudor Achim, and Pieter Abbeel. Bigbird: A large-scale 3d database of object instances. pages 509–516, 05 2014. doi: 10.1109/ICRA.2014.6906903.
- Sudipta Sinha, Johannes Kopf, Michael Goesele, Daniel Scharstein, and Richard Szeliski. Image-based rendering for scenes with reflections. *ACM Transactions on Graphics - TOG*, 31:1–10, 07 2012. doi: 10.1145/2185520.2185596.
- Arnold W. M. Smeulders, Dung Manh Chu, Rita Cucchiara, Simone Calderara, Afshin Dehghan, and Mubarak Shah. Visual tracking: An experimental survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36:1442–1468, 2014.
- Julius Orion Smith III. Physical audio signal processing. <https://ccrma.stanford.edu/~jos/pasp/>, 2020.
- S. Song, S. Lichtenberg, and J. Xiao. Sun rgb-d: A rgb-d scene understanding benchmark suite. *CVPR*, 2015.
- Shuran Song, Fisher Yu, Andy Zeng, Angel Chang, Manolis Savva, and Thomas Funkhouser. Semantic scene completion from a single depth image. pages 190–198, 07 2017. doi: 10.1109/CVPR.2017.28.
- Sascha Spors, Rudolf Rabenstein, and Norbert Strobel. Joint audio-video object tracking. *Proceedings 2001 International Conference on Image Processing (Cat. No.01CH37205)*, 1:393–396 vol.1, 2001.
- Auston Sterling and Ming C. Lin. Interactive modal sound synthesis using generalized proportional damping. In *I3D '16*, 2016.
- Auston Sterling, Justin Wilson, Sam Lowe, and Ming C. Lin. Isnn: Impact sound neural network for audio-visual object classification. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, *Computer Vision – ECCV 2018*, pages 578–595, Cham, 2018. Springer International Publishing. ISBN 978-3-030-01267-0.
- Ivan E Sutherland. The ultimate display. In *Proceedings of IFIPS Congress (New York City, NY, May 1965)*, vol. 2, pp. 506-508, 1965.
- Ivan E. Sutherland. A head-mounted three dimensional display. In *AFIPS '68 (Fall, part I)*, 1968.
- Thomas L. Szabo. Chapter 12 - nonlinear acoustics and imaging. In Thomas L. Szabo, editor, *Diagnostic Ultrasound Imaging: Inside Out (Second Edition)*, pages 501 – 563. Academic Press, Boston, second edition edition, 2014. ISBN 978-0-12-396487-8. doi: <https://doi.org/10.1016/B978-0-12-396487-8.00012-4>. URL <http://www.sciencedirect.com/science/article/pii/B9780123964878000124>.
- Zhenyu Tang, Nicholas J. Bryan, Dingzeyu Li, Timothy R. Langlois, and Dinesh Manocha. Scene-aware audio rendering via deep acoustic analysis. *IEEE Transactions on Visualization and Computer Graphics*, 26(5):1991–2001, 2020.
- Petri Tanskanen, Kalin Kolev, Lorenz Meier, Federico Camposeco, Olivier Saurer, and Marc Pollefeys. Live metric 3d reconstruction on mobile phones. pages 65–72, 12 2013. doi: 10.1109/ICCV.2013.15.
- Lonny L. Thompson. A review of finite-element methods for time-harmonic acoustics. 2006.
- Nicolas Tsingos, Emmanuel Gallo, and George Drettakis. Perceptual audio rendering of complex virtual environments. *ACM Trans. Graph.*, 23:249–258, 08 2004a. doi: 10.1145/1186562.1015710.

- Nicolas Tsingos, Emmanuel Gallo, and George Drettakis. Perceptual audio rendering of complex virtual environments. *ACM Trans. Graph.*, 23(3):249–258, August 2004b. ISSN 0730-0301. doi: 10.1145/1015706.1015710. URL <http://doi.acm.org/10.1145/1015706.1015710>.
- Jack Valmadre, Luca Bertinetto, João F. Henriques, Andrea Vedaldi, and Philip H. S. Torr. End-to-end representation learning for correlation filter based tracking. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5000–5008, 2017.
- Kees van den Doel and Dinesh K. Pai. The sounds of physical shapes. *Presence*, 7:382–395, 1998.
- Kees van den Doel, Paul Kry, and Dinesh Pai. Foleyautomatic: Physically-based sound effects for interactive simulation and animation. 06 2001. doi: 10.1145/383259.383322.
- Aaron van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. W. Senior, and K. Kavukcuoglu. Wavenet: A generative model for raw audio, 2016a.
- Aäron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew W. Senior, and Koray Kavukcuoglu. Wavenet: A generative model for raw audio. *ArXiv*, abs/1609.03499, 2016b.
- Aäron van den Oord, Nal Kalchbrenner, and Koray Kavukcuoglu. Pixel recurrent neural networks. *CoRR*, abs/1601.06759, 2016c. URL <http://arxiv.org/abs/1601.06759>.
- Paul Voigtlaender, Michael Krause, Aljosa Osep, Jonathon Luiten, Berin Balachandar Gnana Sekar, Andreas Geiger, and Bastian Leibe. Mots: Multi-object tracking and segmentation. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7934–7943, 2019.
- Ulrike von Luxburg. A tutorial on spectral clustering, 2007.
- J. Wang, W. Wang, Y. Huang, L. Wang, and T. Tan. M3: Multimodal memory modelling for video captioning. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7512–7520, 2018.
- John Wang and Edwin Olson. Apriltag 2: Efficient and robust fiducial detection. pages 4193–4198, 10 2016. doi: 10.1109/IROS.2016.7759617.
- Qiang Wang, Li Zhang, Luca Bertinetto, Weiming Hu, and Philip H. S. Torr. Fast online object tracking and segmentation: A unifying approach. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1328–1338, 2019.
- Yuxuan Wang, Deliang Wang, and Ke Hu. Real-time method for implementing deep neural network based speech separation, 2014. URL <https://patents.google.com/patent/US20170061978>.
- Emile Webster and Clive Davies. The use of helmholtz resonance for measuring the volume of liquids and solids. *Sensors (Basel, Switzerland)*, 10:10663–72, 12 2010. doi: 10.3390/s101210663.
- M.J. Westoby, J. Brasington, N.F. Glasser, M.J. Hambrey, and J.M. Reynolds. ‘structure-from-motion’ photogrammetry: A low-cost, effective tool for geoscience applications. *Geomorphology*, 179:300 – 314, 2012. ISSN 0169-555X. doi: <http://dx.doi.org/10.1016/j.geomorph.2012.08.021>. URL <http://www.sciencedirect.com/science/article/pii/S0169555X12004217>.

- Thomas Whelan, Michael Goesele, Steven J. Lovegrove, Julian Straub, Simon Green, Richard Szeliski, Steven Butterfield, Shobhit Verma, and Richard Newcombe. Reconstructing scenes with mirror and glass surfaces. *ACM Trans. Graph.*, 37(4):102:1–102:11, July 2018. ISSN 0730-0301. doi: 10.1145/3197517.3201319. URL <http://doi.acm.org/10.1145/3197517.3201319>.
- Justin Wilson and Ming C. Lin. Avot: Audio-visual object tracking of multiple objects for robotics. 2020a.
- Justin Wilson and Ming C. Lin. Avot: Audio-visual object tracking of multiple objects for robotics. In *ICRA 2020*, 2020b.
- Justin Wilson, Auston Sterling, Nicholas Rewkowski, and Ming C. Lin. Glass half full: sound synthesis for fluid–structure coupling using added mass operator. *The Visual Computer*, 33(6):1039–1048, Jun 2017. ISSN 1432-2315. doi: 10.1007/s00371-017-1383-8. URL <https://doi.org/10.1007/s00371-017-1383-8>.
- Justin Wilson, Auston Sterling, and Ming Lin. Analyzing liquid pouring sequences via audio-visual neural networks. pages 7702–7709, 11 2019a. doi: 10.1109/IROS40897.2019.8968118.
- Justin Wilson, Auston Sterling, and Ming Lin. Analyzing liquid pouring sequences via audio-visual neural networks. pages 7702–7709, 11 2019b. doi: 10.1109/IROS40897.2019.8968118.
- Jiajun Wu, Chengkai Zhang, Tianfan Xue, William T. Freeman, and Joshua B. Tenenbaum. Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling. *CoRR*, abs/1610.07584, 2016. URL <http://arxiv.org/abs/1610.07584>.
- Song S. Khosla A. Yu F. Zhang L. Tang X. Wu, Z. and J. Xiao. 3d shapenets: A deep representation for volumetric shapes. In *CVPR*, 2015.
- Tz-Ying Wu, Juan-Ting Lin, Tsun-Hsuang Wang, Chan-Wei Hu, Juan Carlos Niebles, and Min Sun. Liquid pouring monitoring via rich sensory inputs, 2018.
- Yi Wu, Jongwoo Lim, and Ming-Hsuan Yang. Object tracking benchmark. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37:1834–1848, 2015a.
- Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. pages 1912–1920, 06 2015b. doi: 10.1109/CVPR.2015.7298801.
- J. Xiao, A. Owens, and A. Torralba. Sun3d: A database of big spaces reconstructed using sfm and object labels. *ICCV*, 2013.
- You Xie, Erik Franz, Mengyu Chu, and Nils Thuerey. tempogan: A temporally coherent, volumetric GAN for super-resolution fluid flow. *CoRR*, abs/1801.09710, 2018a. URL <http://arxiv.org/abs/1801.09710>.
- You Xie, Erik Franz, Mengyu Chu, and Nils Thuerey. tempogan: A temporally coherent, volumetric gan for super-resolution fluid flow. *ACM Transactions on Graphics (TOG)*, 37(4):95, 2018b.
- Ning Xu, Linjie Yang, Yuchen Fan, Jianchao Yang, Dingcheng Yue, Yuchen Liang, Brian Price, Scott Cohen, and Thomas Huang. Youtube-vos: Sequence-to-sequence video object segmentation, 2018.
- A. Yamaguchi, C. Atkeson, S. Niekum, and T. Ogasawara. Learning pouring skills from demonstration and practice. *IEEE RAS International Conference on Humanoid Robots*, 2014.

- Akihiko Yamaguchi, Christopher G. Atkeson, and Tsukasa Ogasawara. Pouring skills with planning and learning modeled from human demonstrations. *Int. J. Humanoid Robotics*, 12:1550030:1–1550030:39, 2015.
- Tianyu Yang and Antoni B. Chan. Learning dynamic memory networks for object tracking. *ArXiv*, abs/1803.07268, 2018.
- Yu J. Fan J. Yu, Z. and D. Tao. Multi-modal Factorized Bilinear Pooling with Co-Attention Learning for Visual Question Answering. In *IEEE International Conference on Computer Vision (ICCV)*, 2014.
- Zhou Yu, Jun Yu, Jianping Fan, and Dacheng Tao. Multi-modal factorized bilinear pooling with co-attention learning for visual question answering. *CoRR*, abs/1708.01471, 2017a. URL <http://arxiv.org/abs/1708.01471>.
- Zhou Yu, Jun Yu, Jianping Fan, and Dacheng Tao. Multi-modal factorized bilinear pooling with co-attention learning for visual question answering. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 1839–1848, 2017b.
- Aston Zhang, Zachary C. Lipton, Mu Li, and Alexander J. Smola. *Dive into Deep Learning*. 2020. <https://d2l.ai>.
- Ruo Zhang, Ping-Sing Tsai, J. E. Cryer, and M. Shah. Shape-from-shading: a survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(8):690–706, Aug 1999. ISSN 0162-8828. doi: 10.1109/34.784284.
- Yu Zhang, Mao Ye, Dinesh Manocha, and Ruigang Yang. 3d reconstruction in the presence of glass and mirrors by acoustic and visual fusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PP:1–1, 07 2017a. doi: 10.1109/TPAMI.2017.2723883.
- Zhoutong Zhang, Qiujia Li, Zhengjia Huang, Jiajun Wu, Joshua B. Tenenbaum, and Bill Freeman. Shape and material from sound. In *NIPS*, 2017b.
- Zhoutong Zhang, Jiajun Wu, Qiujia Li, Zhengjia Huang, James Traer, Josh H. McDermott, Joshua B. Tenenbaum, and William T. Freeman. Generative modeling of audible shapes for object perception. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 1260–1269, 2017c.
- Zhoutong Zhang, Jiajun Wu, Qiujia Li, Zhengjia Huang, James Traer, Josh H. McDermott, Joshua B. Tenenbaum, and William T. Freeman. Generative modeling of audible shapes for object perception. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 1260–1269, 2017d.
- Hang Zhao, Chuang Gan, Andrew Rouditchenko, Carl Vondrick, Josh McDermott, and Antonio Torralba. The sound of pixels. In *The European Conference on Computer Vision (ECCV)*, September 2018.
- Changxi Zheng and Doug L. James. Rigid-body fracture sound with precomputed soundbanks. *ACM Trans. Graph.*, 29:69:1–69:13, 2010.
- Changxi Zheng and Doug L. James. Toward high-quality modal contact sound. *ACM Trans. Graph.*, 30(4), July 2011a. ISSN 0730-0301. doi: 10.1145/2010324.1964933. URL <https://doi.org/10.1145/2010324.1964933>.
- Changxi Zheng and Doug L. James. Toward high-quality modal contact sound. *ACM Trans. Graph.*, 30(4): 38:1–38:12, July 2011b. ISSN 0730-0301. doi: 10.1145/2010324.1964933. URL <http://doi.acm.org/10.1145/2010324.1964933>.



Yipin Zhou, Zhaowen Wang, Chen Fang, Trung Bui, and Tamara L. Berg. Visual to sound: Generating natural sound for videos in the wild. *CVPR*, 2018.

Xizhou Zhu, Yujie Wang, Jifeng Dai, Lu Yuan, and Yichen Wei. Flow-guided feature aggregation for video object detection. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 408–417, 2017.

Zheng Zhu, Qiang Wang, Bo Li, Wei Wu, Junjie Yan, and Weiming Hu. Distractor-aware siamese networks for visual object tracking. In *ECCV*, 2018.