ISNN: Impact Sound Neural Network for Audio-Visual Object Classification
Supplemental Document

Auston Sterling[1], Justin Wilson[1], Sam Lowe[1], and Ming C. Lin[1,2]

[1] Department of Computer Science, University of North Carolina at Chapel Hill
{austonst,wilson,samlowe,lin}@cs.unc.edu
[2] Department of Computer Science, University of Maryland, College Park
lin@cs.umd.edu

1 Modal Sound Synthesis

A more detailed description of modal analysis is provided here, but as described in the main paper, please refer to previous work for complete derivations [1,2] Modal sound synthesis can be broken up into two steps: a preprocessing *modal analysis* step to process the inputs and a faster *modal synthesis* step to synthesize individual sounds.

1.1 Modal Analysis

Modal analysis is a process for modeling and understanding the vibrations of objects in response to external forces. We use modal analysis in two ways: as a step in modal synthesis and as a way to model real-world vibrating objects. Vibrations in an object can be modeled with the wave equation [3], but in order to handle arbitrary geometries with unknown analytical solutions, it is more common to perform finite element analysis on a discretized representation of the object [4,1].

Starting with a watertight triangle mesh representation of the object's surface, the interior volume of the object is filled with a tetrahedral volumetric mesh. A finite elements model can then be constructed to represent the free vibrations of the object:

$$M\ddot{r} + C\dot{r} + Kr = 0. \tag{1}$$

For an object with n vertices, $r \in \mathbb{R}^{3n}$ contains the set of displacements from rest position. M, C, and K are the mass, damping, and stiffness matrices, respectively. The mass matrix M (analogous to m in $f = ma$) and stiffness matrix K (analogous to k in $f = kx$) can be constructed from the tetrahedral mesh with the density, Poisson's ratio, and Young's modulus of the object's material.

The damping matrix C models viscous damping, which is not as simple to construct from first principles, so in practice it is approximated as a function of the mass and stiffness matrices. Multiple functions have been applied to damping modeling [5], with one example being the common Rayleigh damping model:

$$C = \alpha_1 M + \alpha_2 K, \tag{2}$$

where α_1 and α_2 are considered material parameters.

Given this representation of a vibrating object, we are interested in determining the frequencies at which it vibrates. This can be accomplished through a generalized eigendecomposition of K and D The resulting eigenvectors can be combined to create a matrix Φ that serves as the transformation between object space and mode space.

Equation 1 can be decoupled by Φ^T into linearly separable *modes of vibration* with solutions of the form:

$$z_i(t) = a_i e^{-d_i t} \cos(\omega_i^d t), \tag{3}$$

where, for a mode i, $z_i(t)$ is the amplitude at time t, a_i is the initial amplitude, d_i is the rate of decay, and ω_i^d is the damped frequency of vibration. ω_i^d and d_i are computed as a result of the eigendecomposition, but please refer to previous work for full derivations [2,5].

This modal analysis step is performed once per object, and is a computationally-intensive task. As a result, the vibrations of any rigid-body object can be decomposed into a sum of damped sinusoids. This applies to both analysis of vibrations in real-world objects and vibrations of synthetic objects. The resulting Φ matrix and each mode's frequency of vibration and damping rate are saved to be used in modal synthesis.

1.2 Modal Synthesis

For real-world objects, impacts make the object vibrate according to its modes of vibration, creating sound waves in the surrounding air. What we hear as the object's impact sound is the sum of all of its modes' vibrations.

For virtual objects, impacts are modeled as an impulse which excites the precomputed modes of vibration. Given an object-space impulse vector $f_o \in \mathbb{R}^{3n}$ corresponding to the displacement vector r, the impulse can be converted to mode impulses f_m with:

$$f_m = \Phi^T f_o \tag{4}$$

These mode impulses set the initial a_i values for each mode i. The sinusoids for the modes are then sampled, starting from the impact time at $t = 0$, and added to produce the final sound. This process can be repeated for different materials, geometries, and hit points to create a set of synthetic impact sounds.

2 3D Scene Reconstruction

2.1 Introduction

In Section 5.4 of the main document, we briefly discuss the application of our method for enhancing 3D scene reconstruction. Existing methods have become very effective at 3D scene reconstruction from RGB-D video, even in real-time. However, due to limitations of vision-based methods, transparent and occluded

objects are still a challenge for these methods. Our utility demonstrates how ISNN can supplement these existing methods, using sound to provide cues about the otherwise-challenging objects. In this document, we provide additional details about the application utility and the algorithm for inserting objects into scenes.

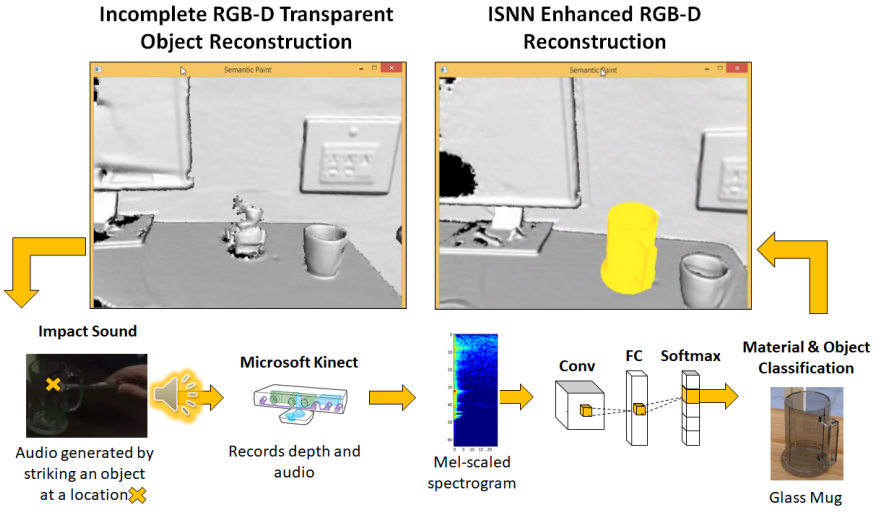


Fig. 1: Our utility begins with a 3D reconstruction of a scene with incorrectly-reconstructed objects. By using ISNN, we are able to complete the scene by inserting the missing geometry. The inserted object is segmented as part of this process.

Figure 1 provides a visual depiction of the algorithm. Please refer to the demo video at http://gamma.cs.unc.edu/ISNN/ for a demonstration and additional results.

2.2 Algorithm

The utility at its simplest provides a system for real-time scene reconstruction, based on previous real-time RGB-D work [6,7]. Using the RGB-D camera of a Kinect, a user scans the scene from multiple angles until estimations have sufficiently converged. At this point, transparent objects may be incomplete or missing. The user interacts with the application to select one of these objects, then physically reaches into the scene to strike the corresponding object.

The Kinect's microphone array records the impact sound, identifies the time of impact, and extracts a 1-second clip containing the sound and its decay. The recorded audio waveform is converted to the form of input to the ISNN-A network: a downsampled mel-scaled spectrogram. This spectrogram is passed

through ISNN-A trained on the full RSAudio dataset. One network trained to perform geometric model classification identifies the closest matching geometry to the recorded sound, while another network trained to perform material classification identifies the closest matching material class.

The full object can then be inserted into the reconstructed scene. The object is inserted at the position earlier selected, using the classified geometric model. In our reconstruction utility, the object is textured with a different color than that of the original geometry, indicating the segmentation of the object from the rest of the scene. Alternatively, the material classification could correspond to a texture which could be applied to the object. As a result of this process, the transparent object which had previously been incomplete or missing, has been both completed and segmented.

2.3 Limitations

ISNN's geometric model classification cannot interpolate or extrapolate geometry given new sounds. When ISNN is trained on the RSAudio dataset, each individual geometric model is considered to be its own class, and classification of a test sound is selection of the closest *training* geometry to that sound. For the utility, this means that the inserted geometric model may be similar to the ground truth object, but not match exactly. Shape optimization from sound is still an open area of research. We have also tested pose estimation methods based on RGB [8] and RGB-D [9,10]; however, future work is needed to extend these to accept asymmetric transparent objects as input and integrate into our application.

3 Feature Analysis

Figure 2 contains a scatter plot of material classes on the axes of the first two principal components. The first principal component explains much of the variation between material classes, as there is clear horizontal delineation—albeit with overlap. This is consistent with the expectation of damping as a material-dependent property. The presence of specific frequency bins that comprise the second component likely delineates model more than material.

4 Additional RSAudio Details

RSAudio contains both real and synthetic impact sounds. A number of the geometric models used to synthesize the sounds are shown in Figure 3. 62 400 synthesized sounds come from a set of 59 geometric models and 11 sets of material parameters categorized into 6 classes of materials. For each model and material pairing (with a few exceptions), 100 sounds with random hit points were synthesized (subsection 1.1).

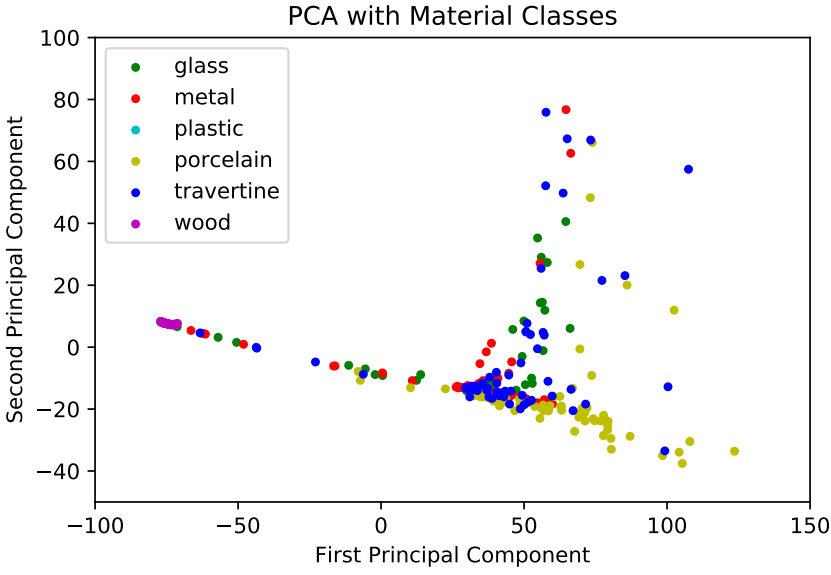


Fig. 2: A scatter plot of material classes on the first two principle component axes. While the horizontal delineation of materials is useful in characterizing those sounds, a full understanding of the relationships between materials and models necessitates a deeper classification scheme.

1183 real impact sounds come from a set of 24 struck rigid objects. These objects are each made of one homogeneous material and primarily consist of dining dishes, utensils, tools, and material samples used for building construction. A majority of the sounds were recorded in a padded sound booth using a Zoom H4 microphone to reduce background noise and room acoustics. The remaining sounds were recorded in a wider set of environments ranging from small offices to large outdoor areas. Each sound contains one impact in isolation from other impacts.

Objects were either struck with a small metal wrench or a rubber-headed drumstick, and in most cases, both. In either case, the striking tool was tightly gripped in a hand while striking in order to minimize its vibrations while the main struck object could vibrate freely. No post-processing was performed to attempt to remove the remaining sound from the striking tool.

5 Additional ModelNet Details

ModelNet10 and ModelNet40 contain virtual objects categorized into 10 and 40 named classes, respectively. This is in contrast to RSAudio's more diverse set

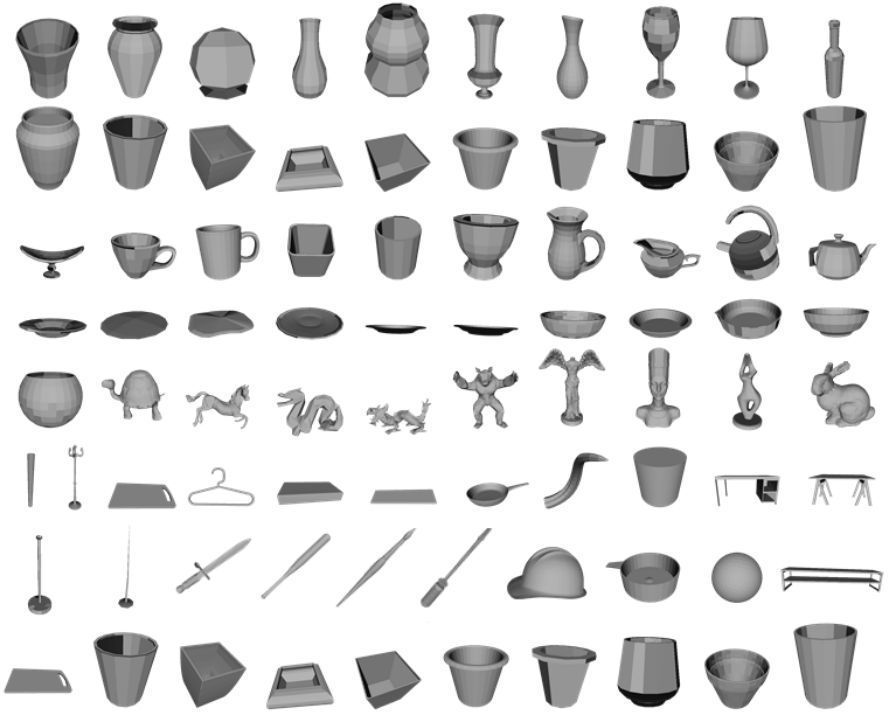


Fig. 3: The RSAudio dataset consists of real and synthesized impact sounds, and is used for training and testing our neural networks for material and object classification. The dataset contains objects of varying shapes and sizes, and are used to synthesize thousands of impact sounds.

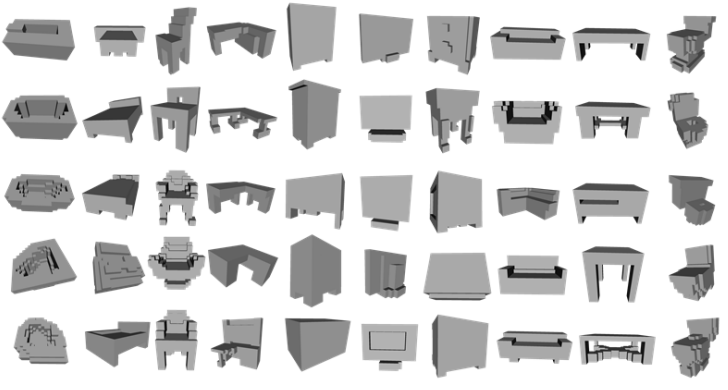


Fig. 4: The ModelNet10 (samples pictured above) and ModelNet40 datasets are collections of virtual objects categorized into fixed numbers of classes.

of small and large objects. We use voxelized representations of these objects as input to our multimodal networks.

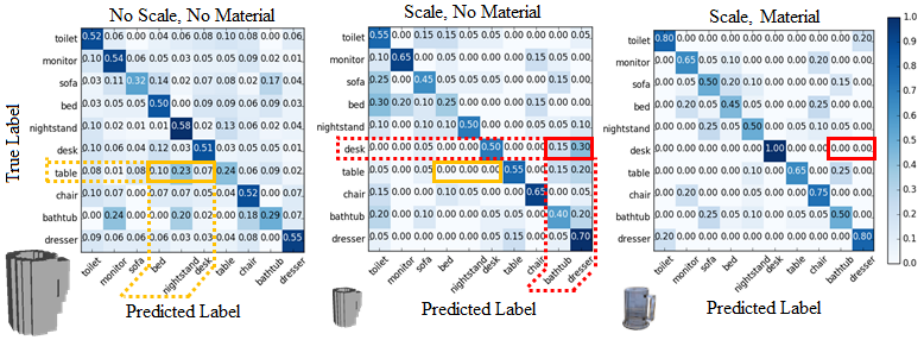


Fig. 5: Confusion matrices on ModelNet10 models scaled and/or assigned a material using ISNN. Applying per-class scaling reduces misclassification between objects of similar geometries but different sizes (yellow), while assigning materials reduces misclassification between objects of similar geometries but different materials (red).

When synthesizing sound with ModelNet objects, by default we use the full voxelized shapes. We also consider the effects of two modifications to the synthesis process: per-class scaling and realistic material assignments. For per-class scaling, each ModelNet class is assigned a scale, so that, for example, monitors are smaller than dressers, which are smaller than beds. Sound synthesis is then performed on the scaled version of the object. For realistic material assignments, each class is assigned an appropriate material for synthesis instead of synthesizing sounds for all possible combinations. Figure 5 shows the differences in resulting confusion matrices when applying these two modifications. Adding auditory data reduces the ambiguity as is shown by the increasing accuracy overall and in the red/yellow highlighted regions from left to right.

Figure 6 further visualizes the per-class material assignment modification. For each of the model classes in the outer ring of the diagram, the middle ring indicates the selected per-class material, and the inner ring indicates which dataset(s) the class is a part of. While many classes are assigned the "wood" material, there are sufficient other materials assigned that classification accuracy is improved.

6 Material Classification

The audio-only ISNN network can also be trained for the task of material classification. That is, given an input impact sound, ISNN trained in this way will produce an estimate of the material class of the object. This is a task which has been more thoroughly evaluated by previous work, but we are still interested in the performance of ISNN on this same task.

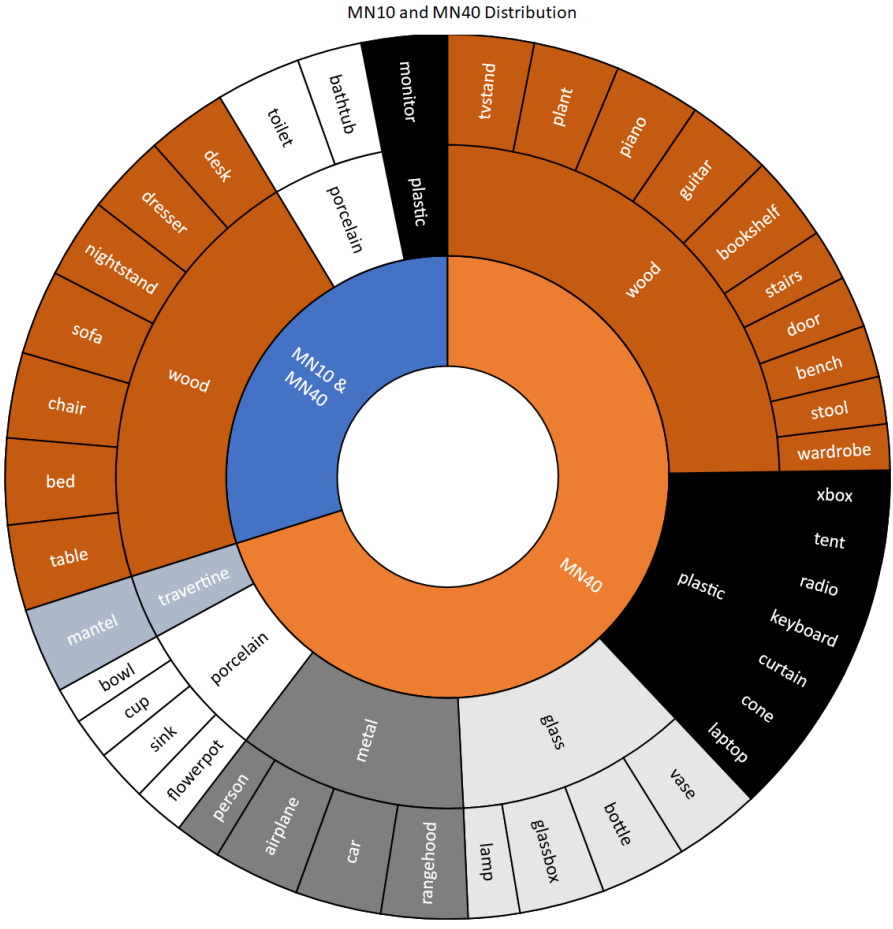


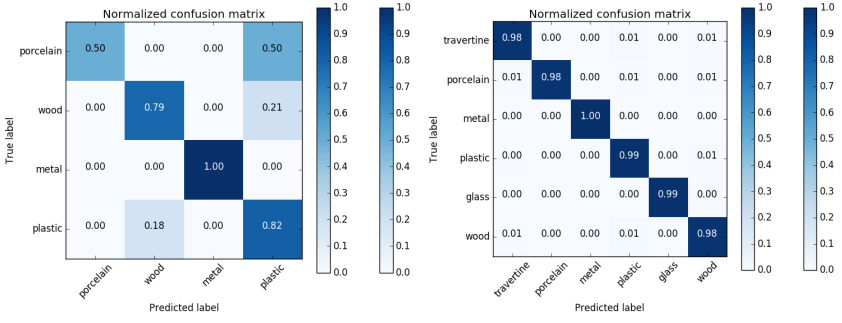
Fig. 6: Sunburst chart showing which model classes are present in ModelNet10 and ModelNet40 (inner and outer rings). Also shows which per-class materials are assigned to each class when performing that modification for sound synthesis.

Material Classification Accuracy

Model	RSAudio S	RSAudio R	Arnab Audio [11]
ISNN-A	**98.69%**	**95.76%**	**71.86%**
SoundNet5 [12]	**99.97%**	29.66%	43.11%
SoundNet8 [12]	92.66%	30.51%	43.11%

Table 1: Material classification accuracy and confusion matrix on subsets of the RSAudio dataset

In Table 1, we compare the material classification accuracy of various classification models on multiple datasets. ISNN-A produces consistently high accuracy, up to 98.69 %, and is either competitive with or or outperforms SoundNet. The material labels provided with the Arnab dataset are not consistent with those listed in their publication, but we selected a subset of those labels with clearly-distinct material names for this test. In comparison to geometry classification (see the main document Table 1), material classification accuracies are a few percent higher on the RSA datasets, but somewhat lower on the Arnab dataset, likely due to the labeling discrepancies.



(a) (b)

Fig. 7: Material classification confusion matrices produced by ISNN-A on (a) the Arnab audio dataset and (b) the synthetic subset of RSAudio. In both cases, there is high accuracy with only a minimal amount of confusion.

In Figure 7, we look at a breakdown of the classifications performed by ISNN on RSAudio's synthetic sounds and Arnab sounds. While RSAudio produces consistently accurate classifications with only minor error, the majority of misclassifications on the Arnab dataset come from porcelain classified as plastic.

7 Activation Maximization

We additionally use activation maximization to visualize the spectrogram inputs which would produce the highest activation for a given ModelNet class. Figure 8 shows how the result of activation maximization changes as different modifications to ModelNet sounds are performed. When no scale or material are applied, the maximized spectrogram demonstrates a need for robustness to variance in frequency and damping. When scale is fixed, so is the fundamental frequency, as can be seen by the single active region and lower overall activation weights. When material is fixed, so are the damping rates, which become recognizable identifiers for this particular class of object.

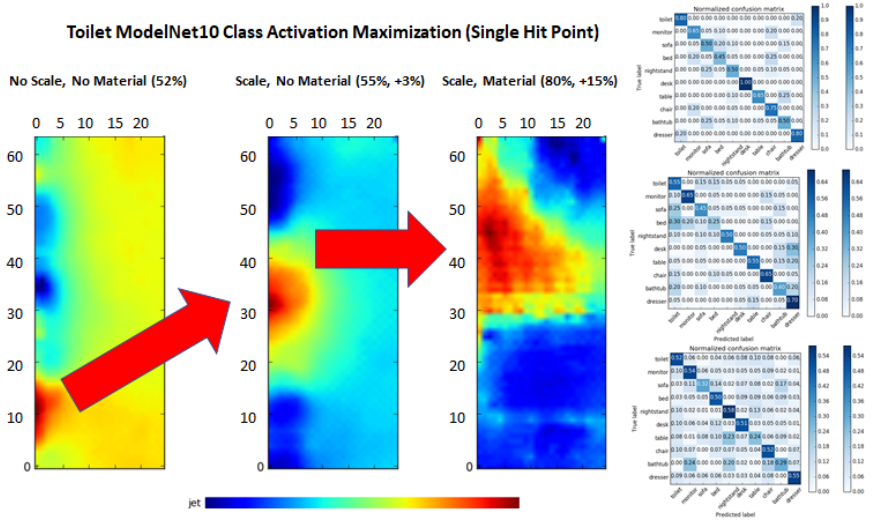


Fig. 8: Activation maximization results for the Toilet class of ModelNet10 as different modifications are made to the model for sound synthesis. When both scale and material are not fixed as distinguishing factors, the network must be general and robust to differences (left). When both are fixed, the network clearly identifies a recognizable pattern (right).

References

1. O'Brien, J.F., Shen, C., Gatchalian, C.M.: Synthesizing sounds from rigid-body simulations. In: Proceedings of the 2002 ACM SIGGRAPH/Eurographics Symposium on Computer Animation. SCA '02, New York, NY, USA, ACM (2002) 175–181 1
2. Raghuvanshi, N., Lin, M.C.: Interactive sound synthesis for large scale environments. In: Proceedings of the 2006 Symposium on Interactive 3D Graphics and Games. I3D '06, New York, NY, USA, ACM (2006) 101–108 1, 2
3. van den Doel, K., Pai, D.K.: The sounds of physical shapes. Presence **7** (1996) 382–395 1
4. Morrison, J.D., Adrien, J.M.: Mosaic: A framework for modal synthesis. Computer Music Journal **17**(1) (1993) 45–56 1
5. Sterling, A., Lin, M.C.: Interactive modal sound synthesis using generalized proportional damping. In: Proceedings of the 20th ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games. I3D '16, New York, NY, USA, ACM (2016) 79–86 1, 2
6. Golodetz*, S., Sapienza*, M., Valentin, J.P.C., Vineet, V., Cheng, M.M., Arnab, A., Prisacariu, V.A., Kähler, O., Ren, C.Y., Murray, D.W., Izadi, S., Torr, P.H.S.: 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. 3

7. Valentin, J., Vineet, V., Cheng, M.M., Kim, D., Shotton, J., Kohli, P., Niessner, M., Criminisi, A., Izadi, S., Torr, P.H.S.: SemanticPaint: Interactive 3D Labeling and Learning at your Fingertips. ACM Transactions on Graphics **34**(5) (2015) 3
8. E. Brachmann, F. Michel, A.K.M.Y.Y.S.G.C.R.: Uncertainty-Driven 6D Pose Estimation of Objects and Scenes from a Single RGB Image. Conference on Computer Vision and Pattern Recognition (CVPR) (2016) 4
9. Ilya Lysenkov, V.E., Bradski, G.: Pose Estimation of Rigid Transparent Objects in Transparent Clutter. 2013 IEEE International Conference on Robotics and Automation (ICRA) (2013) 4
10. Lysenkov, I., Eruhimov, V., Bradski, G.: Recognition and pose estimation of rigid transparent objects with a kinect sensor. In: Robotics: Science and Systems Conference (RSS). (2013) 4
11. Arnab, A., Sapienza, M., Golodetz, S., Valentin, J., Miksik, O., Izadi, S., Torr, P.H.S.: Joint object-material category segmentation from audio-visual cues. In: Proceedings of the British Machine Vision Conference (BMVC). (2015) 8
12. Aytar, Y., Vondrick, C., Torralba, A.: Soundnet: Learning sound representations from unlabeled video. In: Advances in Neural Information Processing Systems. (2016) 892–900 8