A Computer Vision Inspired Automatic Acoustic Material Tagging System for Virtual Environments

Mattia Colombo† https://orcid.org/0000-0002-4169-2045
Alan Dolhasz† https://orcid.org/0000-0002-6520-8094
Carlo Harvey† https://orcid.org/0000-0002-4809-1592

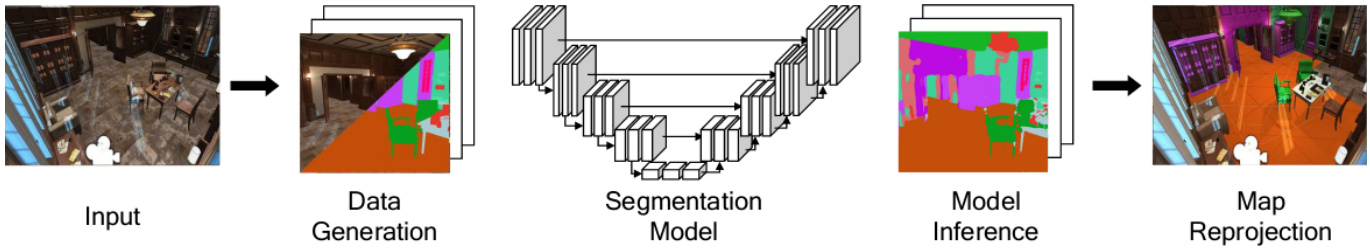

Fig. 1. Our proposed system: given a set of views captured in a VE we perform semantic segmentation using a FCN trained on samples of our scenes. The semantic maps are then reprojected onto objects in the VE. By associating the material classes with acoustic profiles and scene geometry, this information can be used in physically based audio rendering engines.

***Abstract*—This paper presents the ongoing work on an approach to material information retrieval in virtual environments (VEs). Our approach uses convolutional neural networks to classify materials by performing semantic segmentation on images captured in the VE. Class maps obtained are then re-projected onto the environment. We use transfer learning and fine-tune a pre-trained segmentation model on images captured in our VEs. The geometry and semantic information can then be used to create mappings between objects in the VE and acoustic absorption coefficients. This can then be input for physically-based audio renderers, allowing a significant reduction in manual material tagging.**

***Index Terms*—acoustic applications, machine vision, semantic networks, games, rendering (computer graphics)**

I. Introduction

Auditory information is paramount to human perception in natural and virtual environments, helping in orientation and navigation, increasing immersion and aiding in task performance [1]–[3]. The sound field perceived by a listener is a function of shape, dimensions, boundaries and transmission mediums of the surrounding environment. Even though the physics of sound propagation make realistic audio rendering a challenging task, many proposed approaches allow realistic simulations of sound fields in virtual environments (VEs). Computer games, compelling simulations and digital tourism benefit from realistic audio rendering and improved auditory presence evoked in virtual environments [4]–[6]. Modern approaches to audio rendering can be broadly categorised into geometrical acoustics (GA) methods [7] or finite or boundary element methods (FEM/BEM) [8]. Finite elements methods, such as wave-based audio renderers, often require the positions of sound sources and listeners, as well as the scene geometry and associated materials tagged with acoustic absorption coefficients for each material [9], [10]. This process is commonly performed manually, often at significant cost, due to the human-in-the-loop. Our work proposes a first step towards creating an automatic process for generation of such input data for pre-computed audio rendering pipelines, in the absence of knowledge of geometry and material information of a complex scene. Specifically, we propose a proof-of-concept system for vision-based material information retrieval, which allows for near-real-time tagging of an objects' acoustic properties based on its image features, which are then mapped to frequency dependent absorption coefficients. The system tags meshes in VEs representing boundaries in sound propagation paths having noticeable perceptual impact, facilitating the use of GA or FEM/BEM-based acoustic renderers on complex scenes. Our contributions are: 1) a system for material-based tagging of VEs; 2) a methodology for fine-tuning material-based semantic segmentation models; 3) an approach to reprojection of semantic labels back into the VE using level-of-detail.

II. Related Work

This paper leverages recent advances in semantic segmentation, to extract information relevant to realistic audio rendering. Semantic segmentation tasks aim to assign a semantic class label to every pixel in the input image. Examples of applications in scene understanding include PixelNet [11], which performs semantic segmentation and edge detection; EdgeNet [12], which combines depth information with semantic scene completion, using RGB-D input data. For synthetic data generation, UnrealCV provides a pipeline that generates images from VEs providing semantic segmentations [13], allowing for easy generation of training data. However, few examples of applying computer vision to realistic audio rendering exist. One approach [14] uses 360° photographs,

† with DMT Lab, Birmingham City University, UK.

978-1-7281-4533-4/20/$31.00 ©2020 IEEE

TABLE I
THE TABLE DETAILS THE TWO ARCHITECTURES FOR IMAGE SEGMENTATION TASKS TRAINED FOR THE SYSTEM.

Architecture	Backbone	Capacity	F_1-score	IOU	Epochs
Unet	ResNet-34	$24.5M$	0.51	0.58	70
Unet	ResNet-50	$32.6M$	0.54	0.47	70

and depth estimates to generate 3D geometry and semantic information, which is then used for physically-based audio rendering and can also adapt to VEs. In this context, even approximate semantic information could allow for gains in efficiency and decrease in costs of applying physically-based audio rendering to VEs. Modelling sound fields often requires simulations of phenomena such as diffraction and reflection that occur naturally in sound propagation. Recent work on acoustic rendering simulates such phenomena by encoding the sound field using impulse responses (IRs), which decribe spatial information of propagation paths, taking tagged meshes as input. In 3D scenes, efficient acoustic rendering methods includes the Uniform Theory of Diffraction [15] as well as parametric wave field encoders and renderers to simulate such phenomena [10], [16].

III. METHODOLOGY

Based on advances in scene understanding and current state of wave-based audio renderers, we design an architecture that enables the process of semantic mesh labelling in complex scenes and associating every category with a frequency-dependent acoustic absorption function. Methods based on perceptual metrics should consider only meshes that are relevant to the acoustic environment. Scene understanding methods and inference should be optimised depending on the scene geometry.

A. Input

We demonstrate the usage of the pipeline in two scenes: an open space, urban environment (City) and an indoor wooden room (Office). City has $6{:}3M$ triangles and $8{:}6M$ vertices. Office has $3{:}3M$ triangles and $3{:}8M$ vertices. We define a set of classes using tables of measured acoustic absorption of construction materials, where materials are grouped in categories specifying a vector of absorption coefficient values across an approximated equivalent rectangular bandwidth frequency scale ranging from $125Hz$ to $4kHz$. For every major material category that exists in our material database, we define two levels, representing the low and high bounds of mass density in that category. Mass density is a physical property allowing for the acoustic properties of two objects made of the same material to be perceptually distinguishable [17]. We define 23 material classes constituted by the two density levels for each of the 11 material categories and an additional class representing "air", see Table II.

B. Data Generation

We implement the core material tagging system in Unity using a camera located across probe points of a complex scene. Segmentation masks associated with each view are generated

TABLE II
THIS SHOWS THE RANGE OF MATERIAL CLASSES USED IN THE TAGGING PROCESS, COVERING A WIDE RANGE OF ACOUSTIC FILTERS CONVENTIONALLY AVAILABLE FOR AUDIO ENGINES IN GAMES. FILTERS ARE COLOURED BY THE LABEL USED IN FIG. 3, LOW AND HIGH α.

Material	Low α	High α
Glass and glazing		
Masonry walls		
Stud-work & lightweight walls		
Wood & wood panelling		
Floors		
Panels & doors		
Other		
Wall treatments & construction		
Ceilings		
Mineral wool & foams		
Audience & seating		

by ray-casting through each point of C_n, the *near* camera clipping plane, to *1*. For this case, we exclude wavelength-based strides, to maximise segmentation accuracy. The areas where rays intersect with C_f, the *far* camera clipping plane, are labelled as air, objects that are hit by a ray determine the pixel value of the mask which points to the corresponding material. The dataset consists of 3500 labelled images with 512 512 pixel resolution, split into 3000 training images and 500 validation images. In City and Office rendered view images are generated in different regions of the environments. The different regions delimit spaces for the collection of training and validation data. For each delimited region, sets of points are scattered to cover the walkable space. The camera position is interpolated across points in these sets and rotated between 0 and 2 along the azimuth, and between 0 and along the elevation.

C. Semantic Segmentation Model

A convolutional neural network is used to discriminate materials of objects represented in the camera rendered views. This task is performed with pixel-level semantic segmentation using a ResNet-34-based Unet [18]. The ResNet backbone offers a topology that is easy to train and has excellent generalisation performance. It also provides a compromise between accuracy and number of parameters [19]. The model, pre-trained on the ImageNet dataset, is fine-tuned for 70 epochs minimising focal loss [20]. Table I shows information on the networks trained including total number of parameters, F_1-score, intersection-over-union (IOU) and number of epochs.

D. Model Inference

The output of the model is an m n k matrix M, where m and n are the input image resolution and k is the number of classes. For each pixel, the k channels encode a probability distribution across the classes. Per-pixel classes

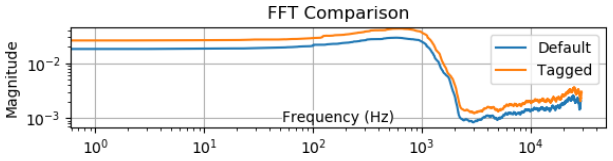

Fig. 2. A comparison between frequency response measured at the RIR probe point.

are determined with the member having the highest presence probability, reducing M to an m n matrix where entries encode the semantic class (see Table II). In addition, counts of unique entries in M determine the number of pixels describing the associated material. With scaling based on the distance between a target object and C_n, this allows material exclusion below a threshold.

E. Map Reprojection

Using the segmented images, meshes are labelled by raycasting through C_n divided in strides. Based on the distance of every Mesh Renderer Unity object inside the camera frustum, the stride size is determined by the lowest structural dimension of each mesh, scaled according to its distance to C_n. This allows consideration of filtering objects by wavelength, $= 0{:}7m$ from the reprojection process. This is because some objects are too small to have significant impact on the human perception of the soundscape [21]. Through this level of detail (LOD) graduation we reduce the analysed scene geometry excluding structures having smaller perceptual impacts on the resulting acoustic model. Among factors affecting performance and accuracy of acoustical simulation methods is the polygon count of the acoustic VE, dependent on the complexity of a scene and the presence of detail and small objects. In acoustic environments, smaller structures on surfaces tend to induce scattering of incidental high-frequency waves reflected, and they are neglected by lower frequencies whose wavelength is greater than the structure dimensions. As a consequence, the amplitude of lower frequencies is more likely to be affected by first order room modes, given by walls and large boundaries, affecting the frequency response of the sound field resulting in a more noticeable perceptual effect. As opposed to frequencies higher than the Schroeder Frequency which tend to scatter chaotically [22], [23]. A pilot study of this perceptual optimisation demonstrates up to 123% of performance gains in offline and real-time acoustic modelling implementations. Small structures on surfaces can therefore be excluded from modelling processes. Results of this process can be seen in Fig. 3 where smaller objects than this of 0.7m do not receive a material tag.

F. Testing

An acoustic renderer is used to test the validity of this method by producing auralisations of Office. City is excluded because of its computationally-expensive scene complexity. We employ a state-of-the-art acoustic renderer [10], integrated into Unity, to generate a model of the acoustic environment in which all meshes having a potential impact on the VE are included. The renderer determines per-mesh absorption information based on the texture meta-data as per Default behaviour. A sound source and listener are placed at human height in the scene; the listener captures a $30s$ chirp signal sweeping logarithmically from $20Hz$ to $20kHz$ emitted by the sound source to measure an IR. Maintaining the same settings and positions of source and listener, we repeat the procedure supplying meshes and absorption information inferred by our system, Tagged. We compare the two IRs generated by the former (Default) and the latter acoustic model (Tagged) through comparisons of their deconvolved frequency responses, see Fig 2.

IV. Results and Discussion

The model inference takes an average of $400ms$ and the reprojection process takes an average of $96ms$. These figures are quoted per camera probe that is used to generate acoustic labels for surfaces in the scene. Images to be inferred are of a fixed size from the scene frame buffer, 512 512 pixels. The time taken for inference is largely invariant to typical scene complexities such as shape, polygon count, materials etc. The Office scene requires 12 probes to completely tag the environment, requiring 6s to complete the tagging process. The City scene shown, has extra complexity and requires use of solutions to the Art Gallery problem to deduce the minimum number of probes to cover the space and tag all objects.

As shown in Fig. 3, acoustic properties can be associated with geometry in the scene, and can be tagged from camera probes. These materials are used in an acoustic rendering process, either directly in game audio engines or external offline acoustic renderers. This can result in more realistic aural spatialisation, using IRs to encode early and late reflections. An example of this offline rendering is shown in Fig. 2. Currently our approach works by providing inference for camera views within the scene. These camera views are manually placed and would need to be placed in many positions in order to tag materials accurately for the entire scene. This process still requires a human-in-the-loop and needs to be addressed to ensure the goal of having this system as an end-to-end autonomous vision based material tagging system. To extrapolate materials tagged to the entire scene, solutions to the Art Gallery problem would optimise the number of predictions required [24], [25]. Considering the polygons encapsulating the walkable space W of a scene, minimum vertex guard algorithms suggest that $bn{=}3c$, where n indicates the total vertices of W, is the least number of positions from where the entire scene can be seen. Based on the depth of the camera, additional intermediate positions $\mathbf{p}$ might be needed to accurately represent objects, this also depends on the number of pixels per object allowing the the neural network to infer materials from the set of camera views that facilitate the whole scene to be visible. For each camera probe position, rotation steps are needed to ensure that all points of W are inside the camera frustum. For an omni-directional camera probe, these rotation steps $\mathbf{r}$ for azimuthal steps and $\mathbf{r}$ for elevation steps should cover the space in 2 azimuth and elevation. The