

SANDS: Spatial Auditory Network Dataset Synthesis

STEVEN DENNIS

BRADLEY LANDRENEAU

CHRIS J. MICHAEL

*Center for Geospatial Sciences Branch
Ocean Sciences Division*

March 29, 2023

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing this collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. **PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.**

1. REPORT DATE (DD-MM-YYYY) 29-03-2023			2. REPORT TYPE NRL Memorandum Report			3. DATES COVERED (From - To) 10/01/2021 – 09/30/2022			
4. TITLE AND SUBTITLE SANDS: Spatial Auditory Network Dataset Synthesis						5a. CONTRACT NUMBER			
						5b. GRANT NUMBER			
						5c. PROGRAM ELEMENT NUMBER			
6. AUTHOR(S) Steven Dennis, Bradley Landreneau, and Chris J. Michael						5d. PROJECT NUMBER			
						5e. TASK NUMBER			
						5f. WORK UNIT NUMBER 1AF4			
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Research Laboratory 4555 Overlook Avenue, SW Washington, DC 20375-5320						8. PERFORMING ORGANIZATION REPORT NUMBER NRL/7340/MR--2023/2			
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) Office of Naval Research One Liberty Center 875 N Randolph Street Arlington, VA 22217-1958						10. SPONSOR / MONITOR'S ACRONYM(S) ONR			
						11. SPONSOR / MONITOR'S REPORT NUMBER(S)			
12. DISTRIBUTION / AVAILABILITY STATEMENT DISTRIBUTION STATEMENT A: Approved for public release; distribution is unlimited.									
13. SUPPLEMENTARY NOTES									
14. ABSTRACT The Spatial Auditory Network Dataset Synthesis, or SANDS, environment provides a technique for rapidly synthesizing auditory network data using modern-day 3D game development libraries. Use of these products bypasses the necessity of building an environmental acoustic model, which would take many years of labor. A combination of the Unity game engine and the Steam audio plugin are used to virtualize sound propagation using ray-traced multipath sound propagation that simulates occlusion, reflection, transmission, scattering, absorption, reverberation, and Doppler effects.									
15. SUBJECT TERMS Spatialized audio Auditory network Acoustic simulation Sound propagation									
16. SECURITY CLASSIFICATION OF:						17. LIMITATION OF ABSTRACT U	18. NUMBER OF PAGES 10	19a. NAME OF RESPONSIBLE PERSON Steven Dennis	
a. REPORT U	b. ABSTRACT U	c. THIS PAGE U			19b. TELEPHONE NUMBER (include area code) (228) 688-4090				

This page intentionally left blank.

CONTENTS

Preface.....	1
Unity Project Preparation.....	2
Project Requirements.....	2
Scene Requirements.....	2
Scenario Definition	4
Global Settings.....	4
Listeners	5
Sound Sources	5
Example Scenario Definition File	7

This page intentionally left blank.

EXECUTIVE SUMMARY

The Spatial Auditory Network Dataset Synthesis, or SANDS, environment provides a technique for rapidly synthesizing auditory network data using modern-day 3D game development libraries. Use of these products bypasses the necessity of building an environmental acoustic model, which would take many years of labor. A combination of the Unity game engine and the Steam audio plugin are used to virtualize sound propagation using ray-traced multipath sound propagation that simulates occlusion, reflection, transmission, scattering, absorption, reverberation, and Doppler effects.

This page intentionally left blank.

PREFACE

The subject of automatic detection and categorization of certain classes of sounds recorded by an auditory network is interesting and useful for several applications ranging from surveillance to mission planning. Modern supervised machine learning techniques are effective in other applications of automatic detection, but require very large amounts of highly curated information to yield favorable results. Unfortunately, no such dataset of curated auditory examples currently exists in the state of the art. In such situations, data synthesis may be used to rapidly create such a dataset without the need of costly field collection, staggering amounts of manual data labeling, and rigorous quality assurance.

The Spatial Auditory Network Dataset Synthesis, or SANDS, environment provides a technique for rapidly synthesizing auditory network data using modern-day 3D game development libraries. Use of these products bypasses the necessity of building an environmental acoustic model, which would take many years of labor. A combination of the Unity game engine and the Steam audio plugin are used to virtualize sound propagation using ray-traced multipath sound propagation that simulates occlusion, reflection, transmission, scattering, absorption, reverberation, and Doppler effects.

Use of SANDS involves the creation of a 3D environment using the Unity engine. Each object within the environment is tagged with its physical attributes as they apply the object's interaction with sound. After the environment is created, the SANDS API may be used to create a scenario. Scenarios are defined using a human-readable text file that specifies the location of listeners, sound sources, tracks of sound sources, and ambiance. SANDS then generates curated output in many forms:

1. An ensemble audio file from each sound source that contains all sound sources of the scenario.
2. An audio file containing the ambient sound devoid of other sound sources during the scenario.
3. A group of audio files, one from each sound source, containing only the respective sound source isolated during the scenario

The SANDS output allows for robust labeling at the time-frequency level at the end-user's discretion. The isolated source output files may be compared against the ensemble in order to determine each source's contribution to the overall soundscape.

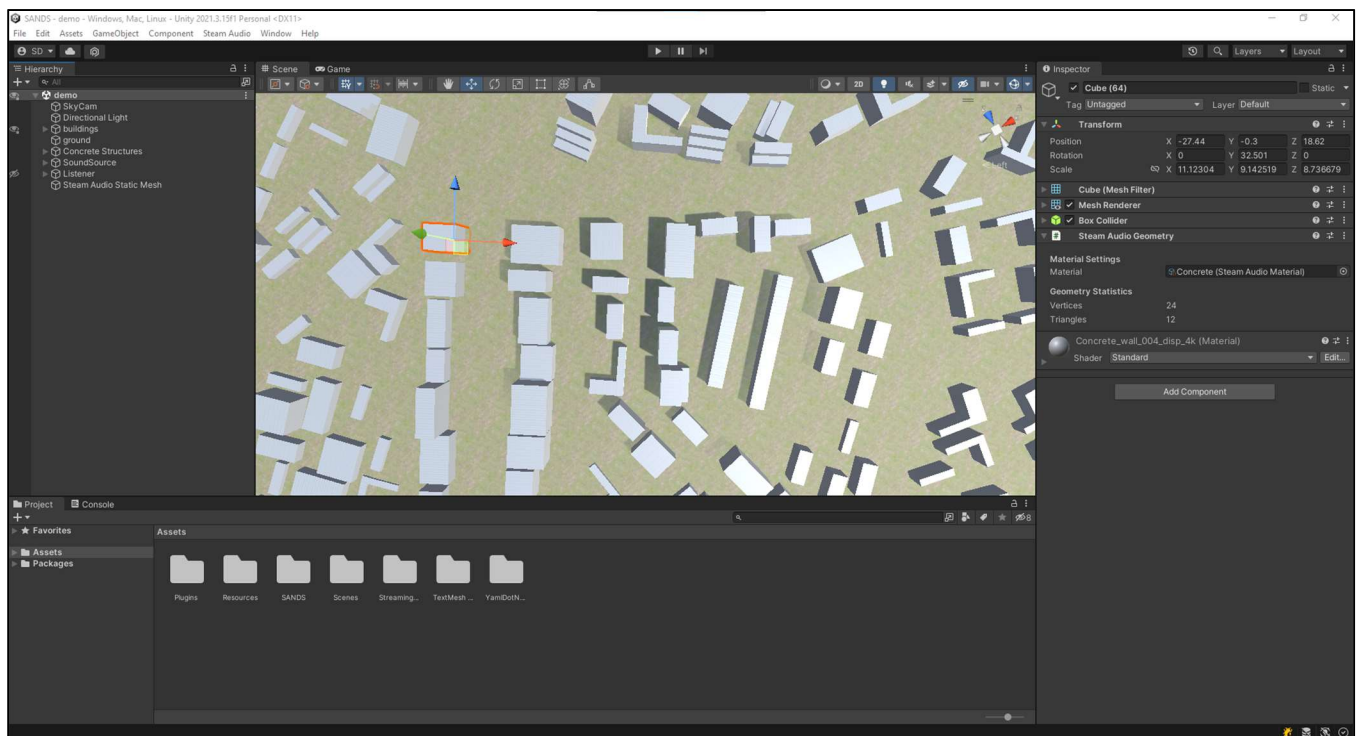
UNITY PROJECT PREPARATION

Project Requirements

The SteamAudio Unity plugin must be installed and enabled for the Unity project. Instructions can be found at the SteamAudio website: <https://valvesoftware.github.io/steam-audio/>.

The YamlDotNet package is required and is available for free from the Unity Asset Store. See <https://assetstore.unity.com/packages/tools/integration/yamldotnet-for-unity-36292> for more details.

The SANDS Unity script files (AudioRenderer.cs, MainController.cs and Scenario.cs) must be present in the project Assets folder (Assets/SANDS recommended).

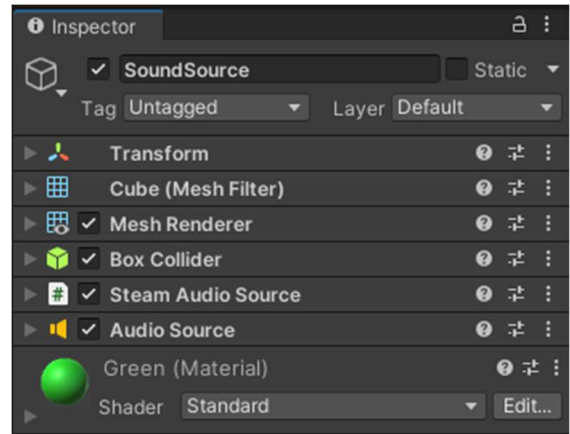


Scene geometry should be tagged with acoustic properties within Unity.

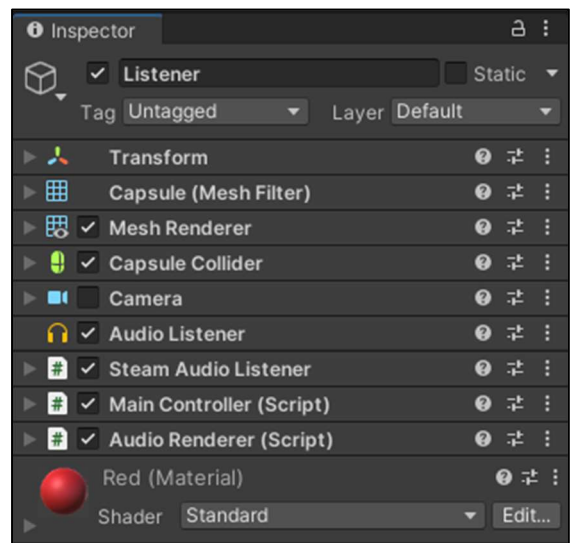
Scene Requirements

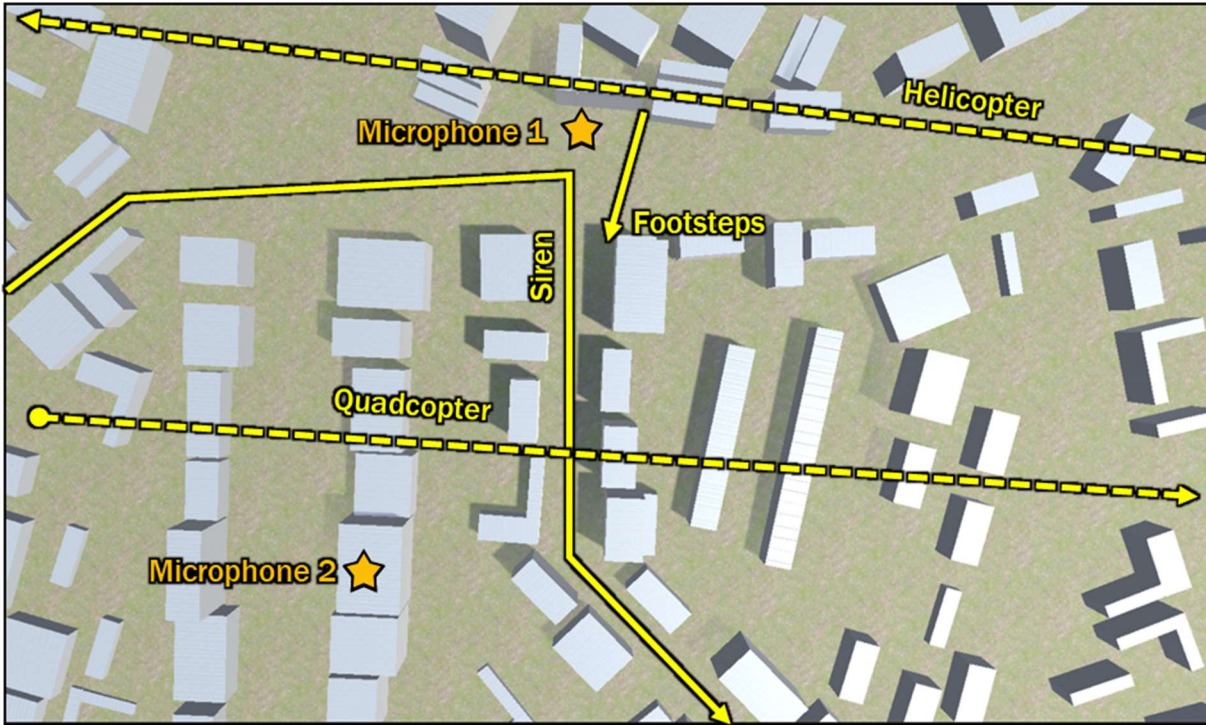
Follow normal SteamAudio procedures for tagging your scene geometry with acoustic properties. Documentation for preparing your scene for SteamAudio in Unity can be found at <https://valvesoftware.github.io/steam-audio/doc/unity/guide.html>.

SANDS requires a single GameObject named **SoundSource** to act as the prototype for all the Sound Sources in a SANDS scenario. Add an Audio Source component to **SoundSource** and configure the settings as desired. There is no need to set an Audio Clip to **SoundSource**. During SANDS synthesis, the scenario audio clips will be assigned for you. The Volume and Spatial Blend parameters will also be automatically set as required by the SANDS scenario. Add a Steam Audio Source component to **SoundSource** and configure the settings as desired for the scenario. You can optionally attach a single child object to **SoundSource** to be used as a visual indicator of when the associated sound is playing.



SANDS requires a single GameObject named **Listener** to act as the prototype Listener in a SANDS scenario. The following components must be added to **Listener**: Audio Listener, Steam Audio Listener, Main Controller (Script), and Audio Renderer (Script). Configure the settings of the Steam Audio Listener component as desired for the scenario.





An example scenario plan indicating sound source paths (yellow) and listener positions (orange).

SCENARIO DEFINITION

Scenario definition is done using a YAML formatted text file. Input fields are in the form of key-value pairs, with input field keys being case sensitive. Standard YAML formatting applies, with new lines indicating the end of a field, indentation with spaces indicating nesting of fields and list elements beginning with a dash. More details on the YAML format can found at <https://yaml.org>.

By default, SANDS will attempt to load a scenario from the file `Assets/StreamingAssets/scenario.yaml`. When launching SANDS from the standalone Unity Player, you can provide a custom path to your scenario input file with the following command line argument:

```
<BuildName>.exe -i path/to/scenario.yaml
```

The following tables detail the input fields and structure of the SANDS scenario file. If omitted, values will take on the noted default values. Values without noted defaults should be considered required.

Global Settings

Key	Value	Notes
<code>IncludeEnsemble</code>	Boolean (default: true)	A value of <i>true</i> synthesizes an ensemble recording of all Sound Sources playing for each Listener. A value of <i>false</i> omits the ensemble synthesis.
<code>SoundsDirectory</code>	String (default: <code>StreamingAssets/sounds</code>)	Path to the folder containing the <code>.wav</code> files for the Sound Sources.

OutputDirectory	String (default: StreamingAssets/output)	Path to the folder where the output .wav files will be written.
PreDelay	Number (default: 0)	Pre/PostDelay sets the amount of silence (in seconds) to include before/after each synthesized recording.
PostDelay	Number (default: 0)	

Listeners

Key	Value	Notes
Listeners	List	A list of Listener elements.
Listener Elements		
Name	String	Identifier for this Listener.
RecordTime	Number (default: 0)	The amount of time (in seconds) to synthesize audio for this Listener.
Position		Defines the position of this Listener in the Unity scene using the coordinates (x, y, z).
x	Number	
y	Number	
z	Number	
Rotation		Defines the orientation of this Listener in the Unity scene using the Euler angle rotations (x, y, z).
x	Number	
y	Number	
z	Number	

Sound Sources

Key	Value	Notes
SoundSources	List	A list of Sound Source elements.
Sound Source Elements		
Name	String	Identifier for this Sound Source. If omitted, the sound filename (see below) will be used.
StartDelay	Number (default: 0)	The amount of time (in seconds) to delay the start of this Sound Source's audio and movement.
Sound	String	The filename, without path or .wav extension, specifying the sound file to play for this Sound Source.
Volume	Number (default: 1)	A value between 0 and 1 that indicated the playback volume of the sound. Values of 0 and 1 indicated silence and full volume respectively.
IsSpatial	Boolean (default: true)	A value of <i>true</i> enables spatial processing of the Sound Source.

A value of *false* disables all spatial processing, with no environmental effects, or effects from the positioning of the Sound Source and the Listener. This is useful for background ambience sounds that are to be considered distant or already spatialized and should be recorded as-is by the Listeners.

Route

The following fields define this Sound Source's motion through the Unity scene.

Speed Number (default: 0)

The pre-defined speed (in Unity units per second) of the Sound Source's movement along this route. This value will be the default speed for any Waypoints that do not explicitly provide their own speed.

IsLoop Boolean (default: false)

A value of *true* Indicates that the Sound Source will travel from the last Waypoint back to the first Waypoint and repeat the Route indefinitely. A value of *false* indicates that The Sound Source should stop at the final Waypoint if it is reached.

Waypoints List

A list of Waypoints defining the Sound Sources movement through the Unity scene.

x Number

y Number

z Number

Defines the position of this Waypoint in the Unity scene using the coordinates (x, y, z).

Speed Number (default: noted)

Defines the speed that the Sound Source will move to the next Waypoint. If not specified, the pre-defined Route speed (see above) will be used.

Example Scenario Definition File

```
### GLOBAL SETTINGS
SoundsDirectory: C:\SANDS\sounds
OutputDirectory: C:\SANDS\output
IncludeEnsemble: true
PreDelay: 1
PostDelay: 1
```

```
### LISTENERS
Listeners:
```

```
- Name: mic1
  RecordTime: 20
  Position:
    x: -86.4
    y: 33.63
    z: -100.7
  Rotation:
    x: 0
    y: -45
    z: 0
- Name: mic2
  RecordTime: 20
  Position:
    x: -161.9
    y: 44.9
    z: -152.6
  Rotation:
    x: 0
    y: 60
    z: 0
```

```
### SOUND SOURCES
```

```
SoundSources:
- Name: ambience
  Sound: desert_ambience
  IsSpatial: false
  Volume: 0.1
- Sound: siren
  Route:
    Speed: 25
    Waypoints:
      - x: -176.9
        y: 23.8
        z: 104.3
      - x: -142.6
        y: 27.8
        z: 41.4
      - x: -95.3
        y: 33.3
        z: -77.2
      - x: -188.5
        y: 32.7
        z: -228.5
      - x: -38.5
        y: 38.88
        z: -315.1
- Sound: helicopter
  Route:
    Speed: 60
    Waypoints:
      - x: 338
        y: 86
        z: -501
      - x: -475
        y: 86
        z: 422
- Sound: quadcopter
  StartDelay: 5
```

```
Volume: 0.8
Route:
  Speed: 50
  Waypoints:
    - x: -261.81
      y: 28.68
      z: -49
    Speed: 10
    - x: -261.81
      y: 86
      z: -49
    - x: 611
      y: 86
      z: -215
- Name: footsteps
  Sound: footsteps_desert_boots_sand
  Volume: 0.05
  Route:
    Speed: 2
    IsLoop: true
    Waypoints:
      - x: -79.2
        y: 32.82
        z: -86.03
      - x: -113.75
        y: 32.83
        z: -115.01
```