ADVANCES ON LOCALIZING SOUND SOURCES ON GEOMETRIES

Jeroen Lanslots[1], Sébastien Paillasseur[2], Saadhana Venkataraman[3], Aswin Ajayan[3], Yash Tailor[3], and Olivier Minck[2]
[1]Siemens Digital Industries Software
Interleuvenlaan 68, Leuven, Belgium
[2]MicrodB, France
[3]Siemens Technology, India

ABSTRACT

Sound Source Localization is more accurate when propagating microphone array sound on 3D geometries rather then on 2D planes. This avoids localization errors for sound sources that are not at all in the same plane. Also, methods that provide quantification results provide much better results when considering the real geometry. Obtaining geometries is not always straight-forward. When a geometry is finally obtained, the microphone array needs to be positioned within the coordinate system of the geometry of the object under test.

Several strategies are possible that will be discussed and compared in this paper. First of all, by manually measuring the translation/rotation between the object under test and the array. This assumes that the geometry is already available, obtained from a CAD file, or is scanned with an industrial scanner. Next, extraction of a geometry based on a single picture. Several methods are possible to do this, e.g. with stereoscopy or with machine learning techniques. As the picture is taken from the microphone array, the translation/rotation is already fixed.

This paper will present results of the above methods and report on effort vs quality of results, comparing source localization results. The test object is an e-powertrain model.

1 INTRODUCTION

Sound Source Localization has a taken a big increase of usage over the last 2 decades, and we saw two disruptors in this market. The first one was the transition of large regular-spaced microphone arrays designed for nearfield acoustic holography (NAH), to the introduction of beamforming arrays. This allowed arrays with less microphones to measure in the farfield and obtain very usable results. And secondly, in the last decade we saw the transition of analog

measurement microphones to digital MEMS microphones. That greatly reduced the cost of arrays, while many more microphones could be placed on the same array area.

On the processing side we also saw spectacular evolutions to improve beamforming results, especially in low frequency. These methods apply NAH on beamforming arrays, use deconvolution methods, or formulate inverse problems with a source-transfer-receiver model. Some of these methods also have the ability to quantify the source. The list is long, a good overview is presented in [1, 2].

One thing has remained though – source localization is most of the times still the back-propagation of the measured acoustic response on a 2D plane. But that contradicts with real-life objects which are not flat, have geometrical shapes, with things sticking out. And, even more importantly: sound sources do not exist in a single plane, but at different distances. Apart from classic beamforming, all sound source localization methods require distance to the source plane. For these methods a compromise has to be taken when selecting a calculation distance.

The solution for this is to use the real geometry of the object. That's easier said then done, because how to easily get such a geometry? This paper describes several methods to get a geometry and discusses its challenges. The object on which this is assessed is an e-powertrain test rig bench which is coupled with an executable digital twin.

2 OBJECT UNDER TEST

For this study we used an electrical powertrain test bench (Fig. 1). This powertrain is coupled with an executable digital twin of the rest of the vehicle running on a real-time platform. It is designed for X-in-the-loop testing, where the 'X' can be 'Hardware', 'Software', or even 'Human'. The mechanical parts are taken from an existing commercial electric vehicle (Kyburz DXP delivery vehicle) and consist of a power invertor, induction motor, a fixed-ratio gearbox with differential, two drive shafts and brakes. An extensive description of this setup and how it interacts with the executable digital twin can be found here [3].

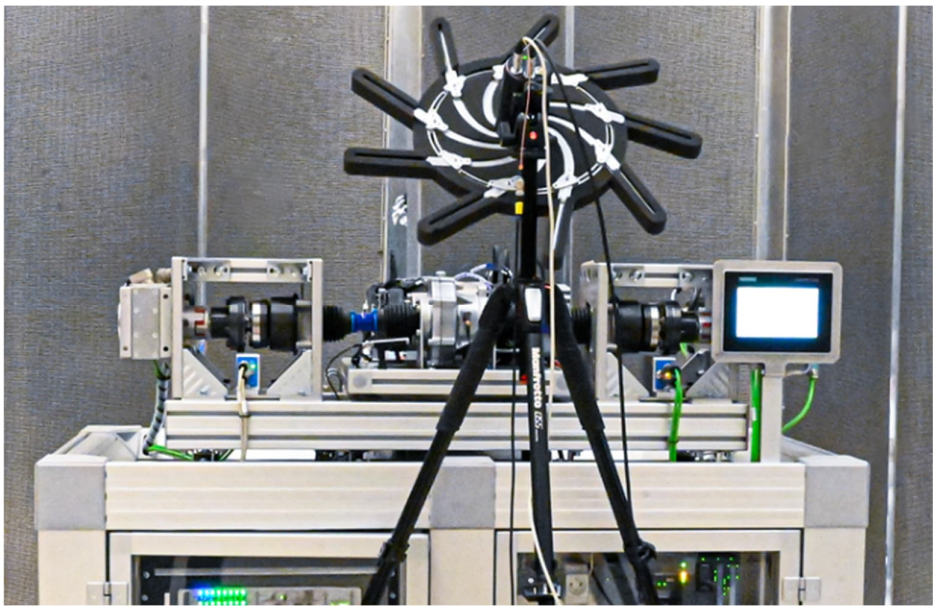

Fig. 1. E-powertrain test rig with real-time machine, controls, microphone array.

The controller of the test bench was set to execute a stepped run-up profile of about 20 seconds. During the run-up profile the rotational speed of the driveshaft increased in steps but stayed constant for a few seconds after each step. The sound was measured with a digital microphone array consisting of 81 microphones and a diameter of about 60cm. Measurements from 3 positions were taken on this e-powertrain test bench (Fig. 2): 45-degree angle, front, and top. The goal is not to do an assessment of this e-powertrain, but to compare results obtained with different geometry methods.

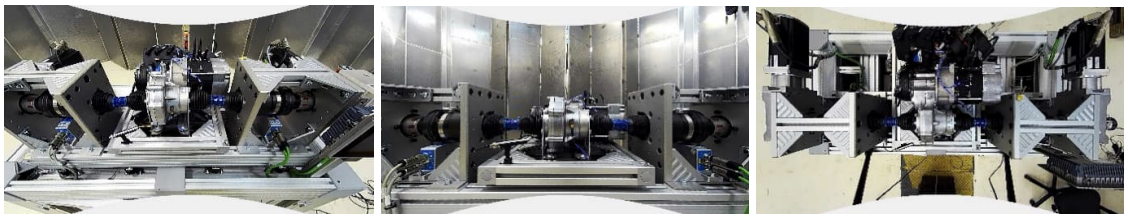

Fig. 2: Array measurement positions with pictures taken by the microphone array.

3 GEOMETRY METHODS

3.1 Geometry from CAD

Getting the geometry from CAD is a convenient way to get a highly detailed geometry of the object under test.

CAD file formats typically do no describe each point in a geometry but use geometrical prototypes such as cylinders, blocks, spheres. For Sound Source Localization we are interested in the outer surface of the object under test. And furthermore, the outer surface should be meshed. That means the surface is replaced by a discrete number of points, a point cloud, and that these points are connected to form triangles. It's these triangles, called elements or faces, that we propagate sound to. Meshing also requires that we omit the interior geometry of components. For example, if we would mesh a CAD file of a gearbox, then we do not need the interior gears to be part of our point cloud. We would not be able to propagate sound on it, and it would unnecessarily increase the calculation. Fig. 3 shows the CAD representation of our e-powertrain test rig. Fig. 4. shows a zoom of detail on the left, and while Fig. 5 shows the meshing of that same detail.

For Sound Source Localization methods that can deliver sound power results, we also like to have the size (or surface area) of these elements as equal as possible. That makes it possible to derive the sound power contribution of a component by summing the contributing elements that form that component. See [4] for an example study. In Fig. 4 it can be seen that the mesh elements do not all have equal size. So, a remeshing would still have to be done to do that.

Finally, also the size and number of the mesh elements is important. First, the size of the elements should be proportional to the highest frequency of interest. There are many rules of thumb when dealing with such 3D geometry meshes, and these depend on the application. For simulation application it is often recommended to have mesh elements at a quarter of the wave length of the highest frequency of interest. For Sound Source Localization however it is sufficient to have mesh elements at about the wave length of the highest frequency.

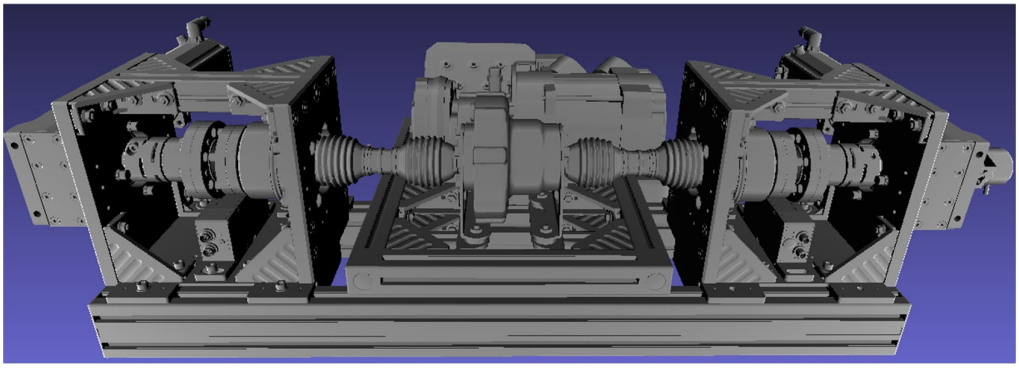

Fig. 3: Full CAD file of the e-powertain test bench

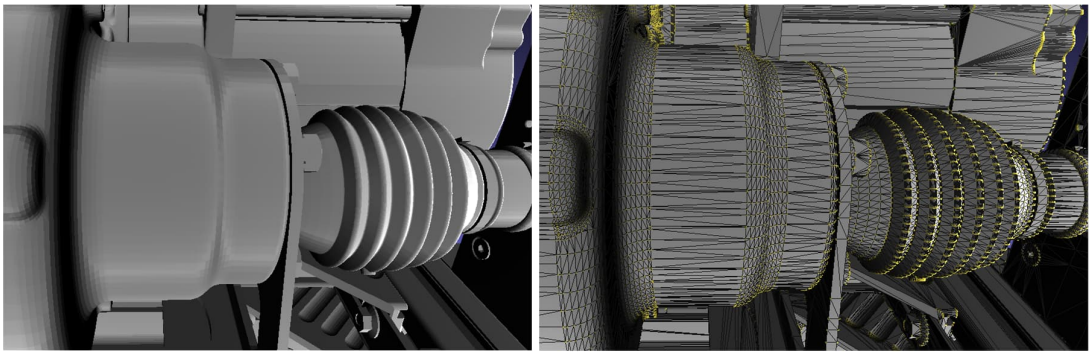

Fig. 4: Detail of the CAD file (left) and meshing into point cloud and elements.

Next, the number of elements is also important. It would be a large overshoot if there would be too many elements in the mesh. For example, the mesh in Fig. 3 and 4 has over 800,000 points, and more than 1.6 million elements. That would seriously compromise the backpropagation of sound to each of these elements. It would require special calculation functions, and moreover, it would not be possible to calculate results in quasi real-time over a full frequency band of let's say 100Hz to 10-20 kHz.

For this analysis our object under test was reduced to 50,000 elements and 24,000 points. The visual representation looks less good, with some details lost, but for sound source localization more than sufficient. With smaller objects, more details would be preserved.

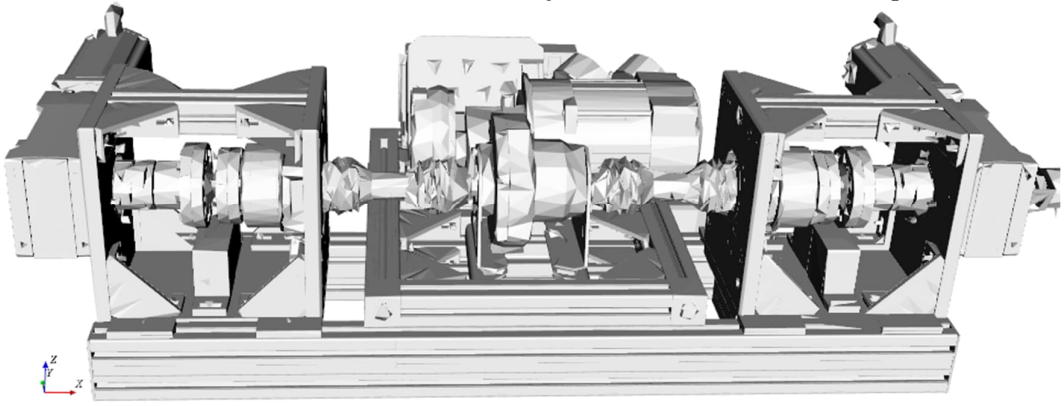

Fig. 5: Remeshing to 50,000 elements.

There are many file formats for CADs, some proprietary such as Siemens NX, others neutral such as STEP. For meshes the STL file format is a commonly used format, and also used for the purpose of this paper.

3.2 Geometry from Scanning

Getting a CAD file is often easier said then done. The right people need to be found in simulation departments, and then the right components have to be carved out, holes need to be closed, it needs to be remeshed. That can take quite some time, days instead of minutes. Often it is much easier to scan the object yourself.

There is a large range of commercial scanning tools on the market, offered in quite a wide price range of about 5k€ to 50k€ and more. Entry level scanners will have limited functionality, take no picture, and have low resolution scans. Entry and mid-segment scanners are typically very sensitive to drift. Drift means that when you move the scanner around the object there is a position shift with respect to the previously scanned part. High-end scanners will have support for detecting the position of the microphone array within the scanned geometry. That makes scanning expensive in term of time spent and/or cost.

What most of these commercial solutions do have is that thy typically come with a complete software package with a lot of functionalities such as: closing holes in the scanner geometry, automatic cleaning, (re)meshing functions, and special processing to deal with drift.

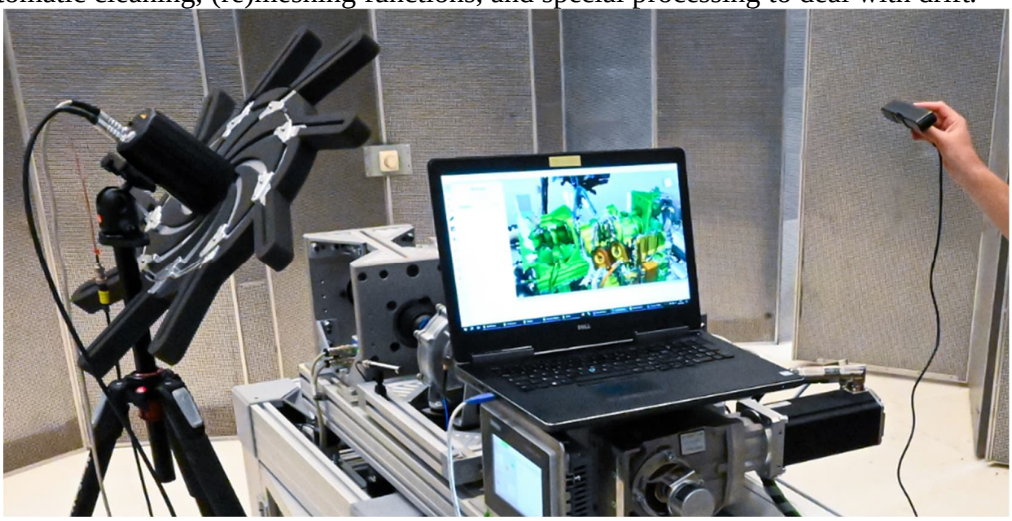

Fig. 6: Scanning the geometry with a handheld depth sensor. (Laptop just for reference.)

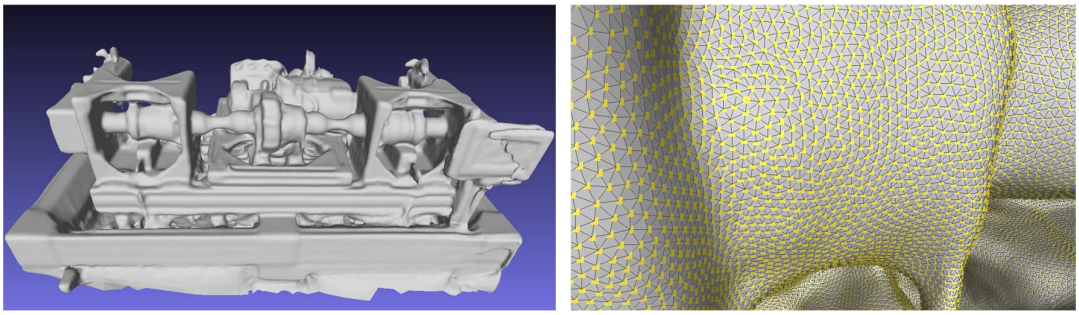

Fig. 7: Scanned geometry (left), and detail of the isotropic remesh (right) where all elements have the same shape and size.

In the example shown in Fig. 6, the geometry was scanned with a handheld depth sensor. That resulted in a point cloud. After meshing the resulting geometry can be seen in Fig. 7 (left). It can be seen that some drift was accumulated during the scan. The whole process took about 15 minutes from scan to the final mesh. The mesh is similar in number of points and elements as the CAD file: 880k points, and 1.7M elements. It was remeshed to about 50k elements. A detail Fig. 7 (right) shows that the remeshing resulted in elements with equal size: it was done with a so-called isotropic remeshing method.

3.3 Geometry from single picture

It's also possible to obtain the geometry from a single picture taken by the microphone array. This paper discusses a machine learning method that can convert a 2D picture into a 3D geometry. Advantage is that the position of the microphone array with respect to the geometry is fixed, as we just have a single picture. That also makes it very fast and moreover, the 3D geometry extraction can always be added later on. Disadvantage is of course that the geometry is only available from a single point of view, and that it could not use observations from a different angle to get a better reconstruction of the geometry.

The method used here uses a deep learning model. A depth image is predicted from a monocular image using a Metric depth model that use a Visual Transformer [5] as the backend. It was trained with several open test datasets, the Stanford 2D-3D-Semantics Dataset (2D-3D-S) [6] and NYU Depth Dataset V2 [7]. These datasets consist labelled data of common indoor scenes of rooms, hallways, floors, walls, and objects in these scenes such chairs, tables, cups, and beds. Given that these scenes are quite far away from mechanical objects with rotating parts that make sound, the geometries that can be obtained are still remarkably good.

Fig. 8 shows the process. The picture taken from the microphone array is run through the neural network. Every pixel has a RGB color, and as an output a 4th parameter is added to each pixel, representing the Depth. So, the picture is converted into a RGBD image. The middle picture in Fig. 8 shows the depth parameter visualized with a color (light/yellow is further away, dark/purple is closer by). Fig. 9 shows the result for the e-powertrain test rig.

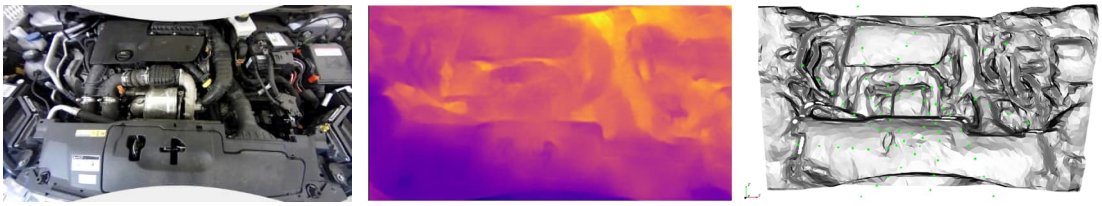

Fig. 8: Geometry from single picture (left), depth image (middle) and resulting geometry (right).

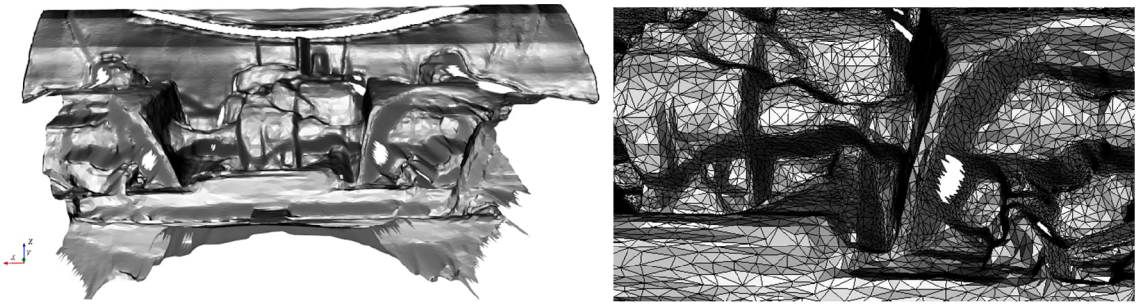

Fig 9: Mesh of the e-powertain test rig (left) and detailed view of isotropic elements (right).

4 USING GEOMETRIES FOR SOUND SOURCE LOCALIZATION

When using geometries for Sound Source Localization, there is one important final step that needs to be addressed. It's the accurate positioning of the microphone array within the geometry. That positioning corresponds to a translation and rotation of the array within the coordinate system of the geometry.

For the geometry extracted from the single picture the position of the array is fixed. We can benefit even more from the measured distance to correctly scale the geometry model.

For high-end scanners, the position of the microphone can be determined with the scanning tool itself.

For both the CAD-based and most other scanned geometries, a straight-forward procedure is to select a number of points on the geometry, a number of points in the microphone array, and then manually measure the distance between each of pair of points.

The below example shows how this was done for the scanned geometry of section 3.2, where a matrix of 4-by-4 points is measured. The 4 points of the microphone array were actually microphones. The 4 points on the e-powertrain can be seen in Fig. 10. The points should be visible from the viewpoint of the microphone array, and best in different planes.

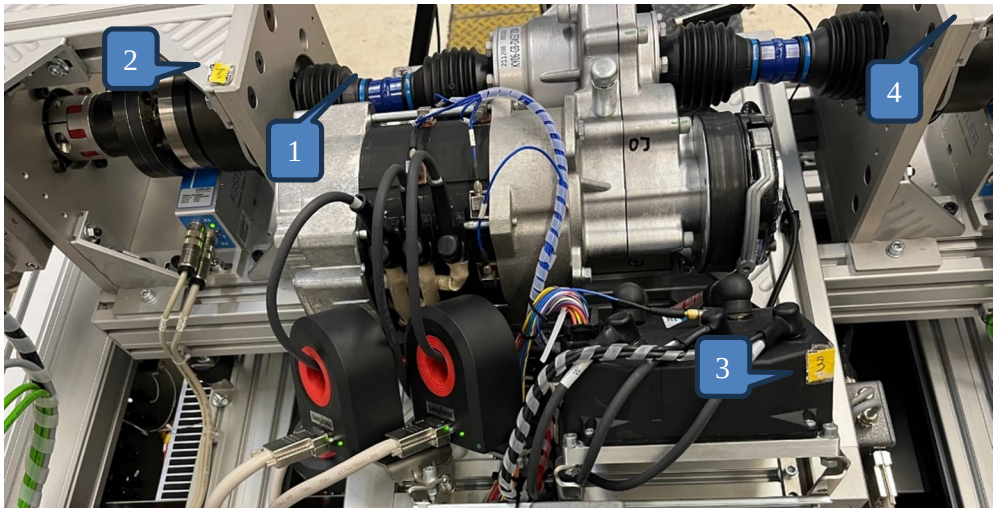

Fig. 10: Selection of points on the e-powertrain test bench.

		Mic1	Mic2	Mic3	Mic4
Mic number		74	77	79	81
	Mesh	Mesh Coordinate	Mic1	Mic2	Mic3	Mic4
Add	1	(0.288 , 0.073 , -0.031)	0.586	0.514	0.641	0.686
Add	2	(0.087 , 0.312 , -0.152)	0.771	0.582	0.543	0.702
Add	3	(-0.345 , 0.194 , 0.109)	0.913	0.964	0.718	0.742
Add	4	(-0.034 , -0.069 , 0.648)	0.692	1.071	0.989	0.695
Add	5	(0.000 , 0.000 , 0.000)	0	0	0	0

Compute

Array position

TX	0.46	RX	-59.0
TY	0.48	RY	39.0
TZ	0.26	RZ	54.9

Fig. 11: Measured matrix of points (left) and calculated array position (right).

Fig. 11 shows the measured matrix, a simple script that calculates the translation and rotation of the array that is then further used for the sound source localization calculations.

The total procedure took about 10-15 minutes per array position.

Finally, let's have a look at the different sound source localization results. Fig. 12 shows the results of the same measurement but on different geometries. All results are calculated with beamforming using cross-spectral matrix deletion method. In the top left is the original 2D picture taken by the microphone array, with the sound source localization results backpropagated on a 2D plane at 50cm distance. That 50cm corresponds to the distance measured from the array center to the point on the object under test right in front of that.

Moving clockwise we see then first the reduced mesh of the original CAD geometry. It can be observed that the sharp edges of the geometry form boundaries or shadow areas: the center of the array, used as a point of view, determines which elements can and cannot be seen.

On the bottom right is the beamforming result on the mesh created from the 2D picture of the top left. It has been cleaned a bit compared to the initial result that was presented in Fig. 9, with some elements in the front and the background of the mesh removed.

On the bottom left is the mesh that was manually scanned with a low-end depth scanner. The mesh itself is a bit more rounded, especially compared to the CAD file mesh. Mesh elements are better visible and also giving a better mapping of the beamforming results over the mesh.

Looking at this mesh from a different angle will give less information of course. Fig. 13 shows the difference between the scanned mesh and the mesh retrieved from the single picture.

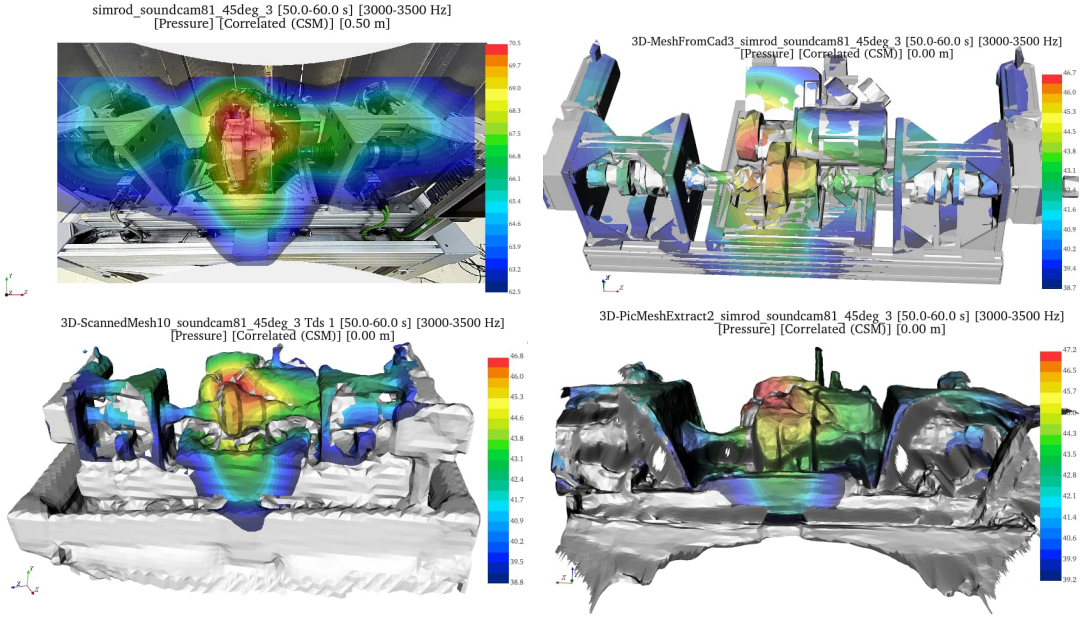

Fig. 12. Clockwise from top left: original 2D array picture, mesh from CAD file, mesh extracted by deep learning model, scanned mesh.

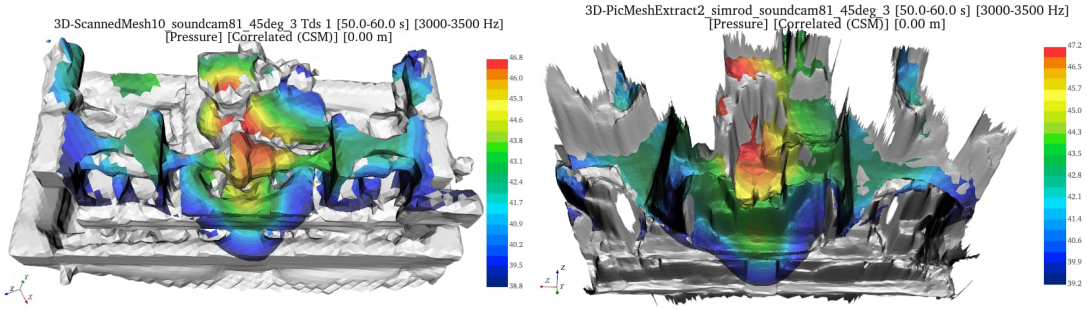

Fig. 13. Full mesh from the top still gives meaningful information compared to extracted mesh.

5 CONCLUSIONS

In this paper, 3D geometries are given as an alternative to the traditional display of sound source locations on 2D pictures with a 2D calculation planes. Sound is then propagated directly on the geometry, with each elements having a different distance to the array. But getting such a geometry and positioning the microphone array within that geometry may not be a simple task. An overview was given on using geometries obtained from CAD models, geometries that are manually scanned, and geometries obtained by machine learning techniques.

CAD models are most accurate but maybe difficult to get from simulation departments, and if even existing, work has to be done to isolate the components used for the test. Scanned geometry models are a good alternative as there is no reliance on external parties, but it requires scanner HW and SW to do the scanning and processing, which is not integrated in array-based sound source localization tools.

In both cases post-processing is required to close holes, meshing to get the surface, reduction to get to a lower number of mesh elements, and finally a remeshing to get mesh elements of equal size. And finally, in both cases, the array needs to be positioned within the coordinate system. Although some scanners support automatically detecting the position of an object (the array), a robust method is proposed based on the point-to-point distance matrix between points on the object under test and points on the microphone array.

CAD-file-based geometries may keep nice sharp edges. Locating sound sources on such a geometry may suffer from these sharp edges creating shadow areas. Adding more microphone array measurements from different angles may improve results there.

Scanned geometries tend to have smoother and rounded shapes. This makes them visually less accurate, but backpropagating sound on such elements result in more usable results.

Extracting a geometry from a 2D picture using deep learning models can be done in less than a minute with a good graphics card. Moreover, it can be done after the measurements during analysis, and only requires the picture and the measured distance. Disadvantage is that it only gives the geometry from a single view angle, mesh reduction still has to be done, and there might be some background elements such as walls requiring an additional clean-up of the mesh.

As expected, the CAD mesh gives the most accurate results, but typically requires the most time to obtain and prepare. The scanned geometry has a very good quality-vs-effort balance: the scanning, preparation and positioning were all done with 30 minutes. Finally, the mesh extracted from the deep learning model is most limited, but given the effort spent it is a very fast method to get a first view on using 3D geometries.

As a recommended future work, it is suggested to look into using separate meshes visualization with a high number of elements. The manual positioning is a robust method that works but has a potential to be replaced by a more advanced method. Finally, the impact of sound power contribution of components of the different geometries should be further studied.

REFERENCES

[1] C. Colangeli. Clustering Inverse Beamforming and multi-domain acoustic imaging approaches for vehicles NVH. Ph.D. thesis. https://iris.univpm.it/bitstream/11566/245537/1/tesi_colangeli.pdf, 2017.

[2] Q. Leclère, A. Pereira, C. Bailly, J. Antoni & C. Picard. A unified formalism for acoustic imaging based on microphone array measurements. International Journal of Aeroacoustics 16 (4-5), 431-456, 2017.

[3] B. Forrier, T. D'hondt *et al*. "A novel E-powertrain test setup for validation and demonstration of multi-attribute and model based testing approaches." Mechatronics Forum International Conference 2023, Leuven, Belgium, 11-12 September, 2023.

[4] S. Paillasseur, C. Chaufour. Measurement of component's sound power emission on a PSA's diesel engine by means of 3D acoustic imaging technique and its applications. 22ème Congrès Français de Mécanique, Lyon, 24-28 August, 2015.

[5] B. Wu, C. Xu, X. Dai, A. Wan, P. Zhang, Z. Yan, … P. Vajda. Visual Transformers: Token-based Image Representation and Processing for Computer Vision. arXiv [Cs.CV]. Retrieved from http://arxiv.org/abs/2006.03677. 2020.

[6] I. Armeni, A. Sax, A.R. Zamir, & S. Savarese. Joint 2D-3D-Semantic Data for Indoor Scene Understanding. ArXiv E-Prints. Retrieved from http://arxiv.org/abs/1702.01105. 2017.

[7] N. Silberman, D. Hoiem, P. Kolhi & R. Fergus. Indoor Segmentation and Support Inference from RGBD Images. ECCV. 2012.