Response to the Reviewer's Comments

May 23, 2017

We thank the reviewers for their insightful and useful feedback. We have revised our paper in light of this feedback, and feel it has benefited greatly from addressing the concerns raised by the reviewers. Please find below our replies and comments about each of the issues raised, and how we modified the paper to address them.

```
>> REVIEWER #1 <<
```

```
> The proposed work simulates underwater imaging sonar based on OpenGL shading
> language (GLSL) chain, and is able to simulate two main types of sonar sensors:
> mechanical scanning imaging sonars (MSIS) and forward-looking sonars (FLS).
>
> The paper is well written with just small issues like "there is any previous
> work for comparison", "foward-looking sonars" or "the sonar simulator can be
> by feature" that needs to be easily solved.
```

Thank you for catching these issues. We corrected the existing typing errors in the revised version.

```
> The contribution of the work is interesting for the community, mainly if the
> simulator will become freely available. It can increase the potential of the
> work in terms of impact.
```

The simulator is available under an open source license and can be downloaded from Github. We added a reference in the outlook section.

```
> About the work, Figure 3 is confused and needs to be improved. The authors
> argue that the beam are composed by the intensity, depth and angular
> distortion matrix, however it is not clean in the figure. The names makes
> the things hard, please "uniform the names", like "Beam Angle in Camera"
> = "Angular Distortion"? "Surface Angle to Camera" = "Intensity"? "Distance
> from Camera" = "Depth" ?
```

We redrew Fig. 3 to make clarify this.

```
> In Fig. 6, it is not clear the effect of the parameter "p". How does the
> normal (blue channel) is adopted in the calculus of the acoustic intensity?
> It is more readable to show the final simulation instead of the normal
> image vs. depth image (green channel), as in Fig. 5.
```

To find the pixel intensity, we multiply its normal angle value by the material reflectance ρ. This operation takes place in fragment shader. We rewrite the lines 258–264 to make it clearer.

> About the simulation resolution, the number of bins are dependent of the
> sonar's frequency and the adopted range. The authors know it, as shown in line
> 266, however it could be interesting to define the number of bins in terms of
> these two parameters to make the simulation "more realistic" in terms of
> the real sensor. It could be interesting for the community.

Jan's answer

> Another interesting thing is related to the noise simulation. It does not appear
> realistic due to the lack of "noise" in the black areas. The application of
> noise in this area can be interesting.

This limitation is explained in lines 376–382. As we described in Section 3.4, the speckle noise modeled in this paper is a multiplicative one, following a non-uniform distribution. The resulting sonar data is composed by an element-wise multiplication between the raw data and the speckle noise. The insertion of additive noise is already addressed as future work to solve this missing part.

> The main limitation of the work is the lacks of reverberation simulation in the
> work. It limits the applicability of the simulation to open waters with a small
> number of objects. Thus, it appears to be the "main future works", however, the
> authors do not mention it.

We have chosen to extend the underwater acoustic phenomena in the simulator obeying its usage by real-time applications. Processing the reverberation in a sonar simulator is computationally costly, then its addition must consider the time consuming again, as we described in the lines 422-426.

> It is hard to validate a simulator, however, I believe the works lake in
> terms of comparison with real data at least in terms of SNR or other metrics.
> It could be interesting to simulate a "real scenario" and compare the results
> obtained by the simulator.

Jan's answer

\>> REVIEWER #2 <<

> The paper proposes a complete simulation suite for real-time underwater
> imaging sonar simulator. It seems to be a very broad proposal, that includes
> many aspects of the simulation itself, going through the rendering to the
> physical simulation itself. Acording to the authors, one main difference of
> their work in relation to others is that they solve both mechanical scanning
> imaging sonar (MSIS) and forward-looking sonar (FLS) models.
>
> The results are promising and interesting. However, I am afraid that the
> contribution is tangent to Computer & Graphics, even with a visualization
> aspect being solved.

Luciano's answer

> The Euclidean distance from camera center, the surface normal angles, and the
> angular distortion are recorded as color channels. In this case, there is a
> limitation of 256 values. Does this brings precision problems? Using CUDA or
> OpenCL, couldn't this be addressed in a more elegant way?

Good points. At the beginning of development, we only had precision limitation problems with the depth information. Since the shader returns data in 8-bit color space, if the number of bins is higher than 256, the distance histogram will contain some blank spaces that will be reflected in the final sonar image as "black holes". To avoid this precision limitation, we store the depth information using the native GLSL depth buffer, which has 32-bit floating-point values.

Indeed, CUDA and OpenCL are both good alternatives ways to implement the proposed approach. However, we have chosen to use GLSL for three main reasons: 1) native from OpenGL, avoiding to install additional packages; 2) hardware and backward compatibility; 3) usage of precomputed geometric informations during the rasterization process.

```
> I would like to see more details on the implementations of the models. I am not
> sure if it is possible to reproduce the solution with the presented text.
```

The simulator is available under an open source license and can be downloaded from Github. We added a reference in the outlook section.

```
> This sentence seems to be from a decade ago paper...: Modern graphics hardware
> presents programmable tasks...
```

Thank you. This sentence has been rewritten in our revised manuscript.

```
> It is unclear to me, but each beam is for a complete column?
```

Each beam is composed by one or more columns, according to sonar bearings. We took this point and revised the text to make it clearer by adding this information in the lines 263–264.

```
>> REVIEWER #3 <<

> The present work proposes a method for the simulation of sonar sensors.
> Differently from previous methods, the proposed technique takes advantage
> of the GPU to achieve realtime performance and is able to reproduce the
> operation of FLS and MSIS sonars sensors.
>
> Text
> ====
>
> The text contains some confusing passages that demand review. For instance:
>
> --- lines 78-82
> "The intensity measured back from the in-
> sonified objects depends on the accumulated energy based on
> surface normal directions, producing more realistic simulated
> scenes, instead of statically defined by the user, as in [5], or in
> a binary representation as found in [6, 7]."
>
> ---> "accumulated energy" refers to what exactly? Energy accumulated where?
> ---> "instead of statically defined by the user" What does it mean?
```

Thank you for catching this point. We have rewritten the subsection 1.1 to avoid such confusion.

```
> --- lines 172-174
> "The Rock-Gazebo integration [17] provides the underwa-
> ter scenario, and allows real-time hardware-in-the-loop simula-
> tions."

> ---> What is "Rock-Gazebo integration"? The authors explain it later. However,
> the order of the sentences could be switched to make text a bit clearer.
```

Jan's answer

```
> The Introduction section contains some text that should be in the Related
> Work Section.
```

Luciano's answer

```
> Technically, bump maps are different from normal maps. Bump maps are defined as
> height maps that describes the bumps on parametric surfaces. Normal maps, on the
> other hand, are textures that contain perturbed normal vectors. There are both
> tangential and object space normal maps. It seems that the proposed technique
> uses normal maps. I suggest a review on that.
```

Thank you for catching this issue. In fact, the proposed approach uses normal maps to simulate geometric surface irregularities by passing RGB textures and modifying the normal vectors. We revised the manuscript to avoid this mistake.

```
> Some image labels are too small.
```

This paper followed the journal's LaTeX template. Unfortunately, we do not think we can change this.

```
> No need to put "the {first|second|third|last} {scenario|scene}" in bold on
> lines 322, 327, 334, 343, 347-348, 354-355.
```

Luciano's answer

```
> Contributions and limitations
> ============================
>
> In the abstract the authors state the importance of simulating sonar sensors
> and then introduce the proposed method. However, nothing is said
> about the difficulties involved in such a simulation and about the general
> limitations of current methods. Along the text, it seems that the use of the GPU
> for sonar sensor simulation is a contribution, but this is not mentioned in the
> abstract.
```

Rômulo's answer

```
> --- lines 69-73
> "This paper introduces a novel imaging sonar simulator, that
> can overcome the main limitations of the existing approaches.
> As opposed to [1, 2, 3, 4, 5, 6, 7], where the proposed models
> simulate a specific sonar type, our model is able to reproduce
> two kind of sonar devices (...)"
>
> ---> These sentences may lead the reader to wrongly interpret the existing
```

```
> techniques, that simulate only one type of sensor, as being limited although
> they could even do it very accurately. Actually, it can be (and probably will
> be) the case that the authors of the existing techniques were not concerned
> about simulating several sonar sensor types. I think that this statements should
> be reviewed and rewritten in order to correctly account for that.
```

Thank you for catching this point. We have rewritten the sentences and the subsection 1.1 to avoid such confusion and make them clearer.

```
> ---> The authors claim that the proposed technique overcomes the "main
> limitations" of existing techniques. However, the proposed method itself
> introduces limitations that were not present in some of the existing techniques
> (for instance, the proposed method does not handle reverberation, an important
> phenomenon in the context of sonar sensor simulation). Thus, I think that the
> discourse about contributions versus limitations should be reviewed in order
> to be fair.
```

Thank you. We addressed your consideration in the text and rewrite the subsection 1.1 to avoid such confusion in the revised paper.

```
> The method
> ==========
>
> The explanation about sonar sensors, and how they work, seems to be Ok. However,
> I've missed a formalization of the problem being simulated. Without it, it
> becomes difficult to understand how good the proposed simulation method is
> (which were the simplifying assumptions adopted?). Also, the description of
> the method implementation is quite confusing. I think that a grad student
> could not implement it without too much guessing.
```

Undefined responsible's answer

```
> --- lines 74-78
> "Also, the underwater scene is processed during the pipeline rendering on
> graphics processing unit (GPU), accelerating the simulation process,
> guaranteeing real-time simulation, in contrast to the methods found in
> [1, 2, 4, 5]."
>
> ---> It is not clear why the GPU/rasterization were chosen and how the GPU
> helps in overcoming eventual problems/limitations of the existing techniques.
> Why GPU-based realtime ray tracing was not chosen, for instance?
```

The rasterization was chosen because the geometric data from virtual scene are computed on optimized pipeline GPU embedded. With this, we handled these pre-computed data to get the depth, intensity and angle properties needed to build the acoustic frame. In addition, we have rewritten the subsection 1.1 to include these informations.

```
> The method takes advantage of the rasterization pipeline in order to build
> the distance, normal and angle maps. This is not stated anywhere, but that
> rendering method is similar to the "deferred shading" (which solves the
> visibility from the camera viewpoint at the same time that stores several
> properties of the rasterized fragments into an auxiliar buffer). This would
> be one advantage related to the use of GPU in this context.
```

Undefined responsible's answer

```
> From Section 2 (lines 104-107) it seems that bins are samples placed along the
> beam. However, I could not spot how these samples are obtained from the
> description in Section 3.3. Actually the text explains it, but I've found it too
> confusing. In the beginning I had the impression that the distance, angle and
> normal maps were 2D. Later on, I started thinking that they should be 3D
> (what does "3D shader matrix" mean in line 257?). However, if they are really
> 3D maps, when is each slice of this 3D map (matrix) obtained? I've got really
> confused at this point.
```

The "3D shader matrix" is a 3-channel matrix which stores the 2D information about depth, intensity and angle from the captured OSG scene. We have rewritten this term on paper to make it clear.

```
> I've found the noise model too arbitrary. How good is this approximation?
```

Undefined responsible's answer

```
> Results
> =======
>
> --- lines 86-88
> "The main goal here is to build quality and low time-con-
> suming acoustic frames, according to underwater sonar image
> formation and operation modes (see Section 2)."
>
> ---> The Results Section does not compare the obtained results with data
> obtained with real sensors (validation). The proposed method is not even
> compared to existing techniques (except for speed in some cases). Thus, despite
> the fact that the proposed method can very efficiently simulate two distinct
> types of sonar sensors, it is not possible to assert how good the results are
> (quality).
```

Jan's answer

```
> --- lines 370-373
> "In real sonar images, the noise also granulates the shadows and blind regions.
> The sonar simulator can be improved by inserting an additive noise to our
> model."
>
> ---> This seems to be a feature that could be easily included. Why it was not
> added to the system?
```

Jan's answer

```
> Here I make an observation regarding the proposed technique. As was
> mentioned in the paper, several existing techniques make use of the ray tracing
> in order to simulate sonar sensors because reflection is extremely
> important in the context of sound propagation simulation and it
> can be easily simulated with ray tracing. Thus, how do the authors
> plan to efficiently expand their rasterization-based system in order to handle
> reflections? Wouldn't it be easier and more efficient with GPU-accelerated ray
> tracing?
```

Good point. We have decided to extend the physical phenomena in the simulator obeying its usage in real-time conditions. Knowing that, our approach is optimized by taking precomputed geometric data during the rasterization pipeline on GPU to compute the acoustic frame instead of simulating sound pulse propagation with ray-tracing approach. Computing the multipath reflection in sonar simulation is computationally costly, thus we need to consider the computation time again by modeling and including phenomena such as reverberation.

```
> --- lines 373-375
> "This lacking part can be addressed by implementation of a multi-path propaga-
> tion model."
>
> ---> What does it mean "multi-path"? References?
```

Multipath means that a signal propagates between transmitter and receiver along several different paths, resulting in fading and reverberation effects. We included the reference [22] in the paper.

```
> Decision
> ========
>
> The proposed work is very interesting but seems to be just in early stages
> of development. Some decisions were not very well justified (why GPUs? why
> rasterization?). Some explanations are confusing (the simulation of the sonars).
> Certainly, with some additional work on these points, the paper will be ready for
> submissions. However, in the current state, and from what was already discussed,
> I've decided for its rejection.
```

We thank for your feedback. We have addressed your comments and the other reviewers' ones in the revised manuscript and we hope your expectations are satisfied this time.

```
>> REVIEWER #4 <<

> The submitted manuscript proposes a sonar simulator for real-time
> applications. Several physical aspects are considered in a computational high
> performance implementation. The final product is impressive.
>
> With respect to the document, it is very well-written and organized, theoretically
> well supported, the presentation is clear and the experiments are sufficient
> to prove the system effectiveness. Everything is well justified and I do not
> have any comment or question about the work. Congratulations.
>
> Under these conditions, I believe the paper is ready to be published.
```

We thank for your feedback. We have addressed the other reviewers' comments in the revised version and we hope your expectations continue to be satisfied.