SoftwareX 30 (2025) 102107

Contents lists available at ScienceDirect

SoftwareX

journal homepage: www.elsevier.com/locate/softx

Original Software Publication

LoDCalculator: A level of detail classification software for 3D models in the Blender environment

Bruno Rodriguez-Garcia, Ines Miguel-Alonso, Henar Guillen-Sanz, Andres Bustillo *

Department of Computer Engineering, Universidad de Burgos, Burgos, Spain

ARTICLE INFO

Keywords:
3D modelling
Level of detail
Virtual reality
Model classification

ABSTRACT

The use of Level of Detail (LoD), a crucial technique in the development of 3D models, implies lower cost graphics and resource economies. These savings are evident in contexts where technical resources are limited, such as immersive Virtual Reality and whenever LoD is critical for accurate representation, such as Cultural Heritage dissemination. Consequently, various systems are used to classify 3D models based on their LoD. However, those systems have several shortcomings that hinder their widespread use. In this research, LoDCalculator, an add-on for Blender open-source modelling software, is presented to address such shortcomings. LoDCalculator ensures unambiguous, universal, and accessible classification of 3D models. It was tested by classifying 12 3D models. The scores were then compared with the evaluations of a group of students and professional 3D modelers in a subjective evaluation. The results of the comparison were satisfactory, showing minimal significant differences between the software and the evaluation group classifications.

Code metadata.

Nr	Code metadata description	Metadata
C1	Current code version	1.0
C2	Permanent link to code/repository used for this code version	https://github.com/DigiUBU/LoDCalculator
C3	Permanent link to reproducible capsule	N/A
C4	Legal Code License	MIT License.
C5	Code versioning system used	Git
C6	Software code languages, tools, and services used	Blender, python
C7	Compilation requirements, operating environments & dependencies	Blender≥ 2.83 LTS
C8	If available Link to developer documentation/manual	https://xrailab.es/cases/lod-calculator/
C9	Support email for questions	labxrai@gmail.com

1. Motivation and significance

The Level of Detail (LoD) concept dates back to 1976 [1], although technological adaptations have over time enabled new advances. It is a feature of great importance in 3D modelling, defining the degree of similarity between an object and its virtual representation [2], and affecting the development and application of 3D models [3]. It is likewise a key factor in 3D modelling projects, as a higher LoD requires more computational and economic resources [4]. The use of LoD is varied on the basis of distance to the user, balancing visual fidelity and performance, so that it is not excessively used [5]. Additionally, determining the LoD of a 3D project considers factors such as the display device and hardware [6].

Proper management of LoD is even more critical in immersive Virtual Reality (iVR) developments for all the above reasons. Projects of that sort where the user is free to inspect 3D models closely will typically require a higher LoD, as noted by several authors [7,8]. This challenge adds to the computational cost of rendering for each eye [9] and the higher LoD required for interactive models [4]. Furthermore, many iVR devices are standalone, meaning the entire rendering process occurs within the iVR device, which has lower power compared to a conventional computer [10]. iVR is a rising technology [11] excelling in multidisciplinary applications such as education, training, and Cultural Heritage (CH) dissemination [12]. In the field of CH dissemination, proper LoD management is also crucial. Such projects typically require maintaining the original structures of objects [13], and avoiding over-simplifications to preserve historical accuracy [14].

Consequently, several popular systems currently classify 3D models based on LoD, such as CityGML and BLOM [4]. Nevertheless, those

* Corresponding author.
E-mail addresses: brunorg@ubu.es (B. Rodriguez-Garcia), imalonso@ubu.es (I. Miguel-Alonso), hguillen@ubu.es (H. Guillen-Sanz), abustilo@ubu.es (A. Bustillo).

https://doi.org/10.1016/j.softx.2025.102107
Received 21 December 2024; Received in revised form 18 February 2025; Accepted 19 February 2025
Available online 27 February 2025

systems have certain limitations that hinder their widespread use. Even though CityGML is the most standardized, there is no total consensus [15]. Some of the most significant limitations are: 1) the ambiguity of classification systems for semantic thresholds, as highlighted in several studies [16–21]. Furthermore, most systems use discrete LoDs, with a series of defined levels [22]. When combined with semantic LoDs, limitations arise when a model cannot be fitted into any of the proposed levels, creating outliers [23]. 2) Lack of universality; those systems are designed for specific contexts, such as CityGML and BLOM, which specialize in urban environments [24]. Their specialization restricts their use in other contexts and complicates comparative analyses. 3) They are not accessible to all audiences, as some require multiple steps and extensive knowledge of 3D modelling [25]. Although the complexity of 3D models has been analysed in other research, those studies have been focused on the differences between versions of a 3D model [26–32], in this regard, Grande's research [33] presents a wide range of techniques and tools for measuring LoD and explores their applications in iVR projects. Moreover, the distribution, installation, and use of those software solutions is quite specialized, making them less accessible to less specialised teams.

1.1. Contribution

This research presents LoDCalculator, a software tool for classifying the LoD of a 3D model, which addresses the previously discussed issues. LoDCalculator is an add-on that works with the open-source 3D modelling software Blender and has been designed to tackle the challenges of threshold definition, universality, and accessibility. LoDCalculator has been tested in CH environment models intended for iVR visualization, for the reasons indicated in the introduction, but it can be used as a general 3D model classification tool, especially in multidisciplinary teams because of its ease of use. Additionally, its open-source nature allows for threshold modifications to improve the classification of other specific types of 3D models.

2. Software description

In this section, an explanation is given of the software design that was developed to overcome the previously mentioned limitations, followed by the operational details of the software.

2.1. Software functionalities

In Table 1 the decisions guiding the design of the LoDCalculator are detailed. Each decision will be explained in more detail in the text, following the same order as in the table, before a full explanation of the software is given. Certain aspects briefly presented in this section, such as the nature of the data, its extraction, and processing, will be further elaborated in Section 2.2 and are showcased in Fig. 2. An important point to clarify that also influenced the design decisions is that the LoDCalculator is a general-use software. It is particularly suited to the analysis of CH models, although it can analyse any type of 3D model that projects realistic visualizations.

Formal thresholds were used to define unambiguous thresholds. However, those thresholds, based on the technical specifications of a model, can also create outliers [34], due to variations in specific features and modelling techniques. A system of formal thresholds based on continuous data from the 3D models was implemented to address that issue. Clear boundaries between each threshold were established, the limits of which were defined on the basis of all the data. So, the first design decision circumvented interpretation errors and results in a total of nine possible scores, ranging from 1 to 5 at half-point intervals. That number of possible LoDs was aligned with existing standards, such as the 10 levels in the GoG [9], the four levels in BLOM3D [4], the five levels in CityGML [19], and the five levels defined by The American Institute of Architects [4].

To address the issue of universality, it is necessary to use relative measures rather than absolute ones, in the same way as the most popular 3D model simplification algorithms do, such as Edge Collapse, Vertex Removal, or Cell Collapse [5]. Although Polygon Count (PC) can indicate the LoD [35], that parameter is not applicable in all cases. For instance, the number of polygons for a single object in a 3D model will in all likelihood be fewer than for an entire city, regardless of the LoD. Moreover, 3D models composed of multiple meshes are expected to lack uniform LoD across all components [36]. So, the following parameters were considered in the analysis. Firstly, geometric fidelity, which refers to the fidelity and surface detail of the 3D model. Secondly, radiometric fidelity, which pertains to accurate reproduction of its visual properties. Both parameters are based on the Münster classification of 3D models for digital reconstruction in the context of humanities research [37]. Lightning parameters were not considered, as that factor was external to the 3D model and varied depending on the game engine. The data used to analyse geometric and radiometric fidelity, respectively, are explained in the following paragraphs.

The software was designed to analyse polygon-based models to ensure the comparison of geometric data, bearing in mind that any parametric model can be converted into a polygon-based one, but not vice-versa. Although those models are usually composed of triangles or squares, the software is specialized in square-based models that are the most common modelling shape [38]. Additionally, more topological differences can be observed in quad-based model tests, making those sorts of models easier to analyse. The decision excludes photogrammetric models, except for those that have been re-topologized, as the mesh of those models is usually generated from triangles [39]. Nonetheless, photogrammetric models are generally characterized by high realism [40].

Considering radiometric properties, a distinction can be drawn between procedural materials and texture-based materials [41]. Texture-based materials utilize images projected onto the 3D model to define its visual properties, whereas procedural materials blend procedural patterns to generate the visual appearance of the 3D model [42]. The software has been designed to analyse texture-based materials, as procedural materials can be converted into textures [43], but not vice-versa. Moreover, this workflow is the most common one when working with game engines [40], particularly in CH projects [43].

On one hand, the accessibility problem is linked to economic factors, as some of the systems of that sort require payment and are based on proprietary 3D-modelling software [2]. As an estimate, some of the most

Table 1
Summary of the design decisions taken to develop the LoDCalculator.

Issue	Threshold ambiguity		Universality of the system			Accesibility of the system	
Question	How to define thresholds	What type of data to use	Measurement type	Geometric data type	Radiometric data type	Distribution	Usability
Solution	9 discrete thresholds with formal boundaries	Continuous formal data extracted from the 3D model	Relative measures based on geometric and radiometric features	Quad-based geometric model analysis (mean angle, PC and vertex concentration)	Texture based material analysis (relative resolution and complexity of materials)	Open-source add-on for open-source software	User-friendly in the Blender interface, only one button and error notification

popular 3D modelling software, such as Cinema 4D, 3D Studio Max, or Maya [44], fall within a price range of €700 to €2000 per annual license, with the latter being the most common price. On the other hand, the systems are often complex to use, especially for individuals who are not fully proficient in 3D modelling, which is common in CH work [45,46], as well as in other multidisciplinary environments related to iVR. The software has been developed as an open-source add-on for Blender, one of the most popular open-source 3D modelling software [2], to address that issue. Usability is enhanced by directly integrating data into the Blender interface with a single interactive button. After installation, a module is added to the scene properties, displaying information on the 3D model, which includes both raw data and the LoD assessment. If any errors are detected, specific notifications appear on an interface panel, guiding the user to rectify the analysis. That feature ensures that novice users can avoid mistakes. Fig. 1 illustrates the interface panel on the right side of the screen showing the results of the Utah Teapot analysis.

Furthermore, its open-source nature permits modifications, enabling adjustments to both the LoD thresholds and the software functionality. This flexibility facilitates analyses of more specific models than those detailed in this research.

2.2. Software architecture

The LoDCalculator was developed in Python using the Blender Python API. The process of acquiring and processing data for geometric and radiometric fidelity is outlined in this section, as well as the procedure for assigning LoD scores. Fig. 2 provides an overview of the software process. Vertically, it depicts the steps from accessing the 3D model data to assigning a score, illustrating how each step impacts the data flow, the process can be divided into the following phases separated in Fig. 2 with the grey dashed rectangles:

1. Acquisition and Processing of Geometry Data: This phase focuses on gathering and processing geometry data to calculate geometric fidelity.
2. Acquisition and Processing of UV Data and Materials: This phase is divided into two sub-steps:
   1.2 UV Mapping Data: Obtaining and processing data associated with the UV mapping of the 3D model.
   2.2 Materials Data: Obtaining and processing data related to the model's materials.
3. Assignment of Scores: once acquired, the data are processed to obtain the final LoD score.

The data used to evaluate geometric fidelity include face angles and the number of faces per mesh in the model. Following a similar approach to Zhao [47], vertices with smaller angles were considered to contribute more information. Low-poly 3D models show higher angle values along their edges, as the large angles provide crucial information. Consequently, a lower number of consecutive steps along a slope indicates a less smooth surface, signifying a lower LoD [38]. When working with relative data, the PC is divided by the number of meshes in the 3D model, yielding the average polygon resolution per mesh. By doing so, comparability between models, whether a few parts of a single mesh or a larger model, is facilitated. The face angles are extracted and normalized to the first quadrant to enable comparison of angles in any direction. Those angles are then divided into quartiles. A higher concentration of smaller angles indicates a more detailed model, as it includes finer details and depicts smoother slopes. This approach allows for the evaluation of geometric surface distortion and detail, similar to previous studies on point cloud surfaces [48]. The degree distance of the last interquartile range is used as a metric: a greater distance from the last interquartile range implies a lower concentration of angles close to 90°, indicating finer details. The average angle within the 3D model was then computed, with higher averages indicating less detail. Lastly, the average number of polygons per mesh was calculated to arrive at the PC.

The mean texture resolution and the proportion of textures per material were chosen as indicators, to evaluate radiometric fidelity, as noted by several authors [39,40]. To do this, the average area occupied by the polygons of the 3D model in the UV space was computed for a relative calculation of texture resolution. The area of each polygon was calculated using a bounding volume circle for area, the simplest and most adaptable procedure, in order to generalize the process [49]. Later, the material nodes were accessed to count the number of materials and texture nodes, and the associated textures were accessed to calculate their mean resolution on one axis. This average area of the UVs was multiplied by the mean texture resolution and the average number of faces per part of the 3D model. In doing so, the mean resolution needed to texture each mesh of the 3D model without overlap could be determined. Additionally, the percentage of polygons with an unwrapped surface area exceeding 100 % of the UV space was calculated, assuming that the textures applied to those polygons were tiled and repeated across their surface area [50].

The data were divided using three thresholds to calculate the LoD score. The data collected in the previous phases were approximated or general, due to generalization; thus, only three thresholds were used to promote coherent segmentation and to avoid outliers. This is one of the system's limitations, as a higher number of thresholds could lead to outliers. However, the selected number aligns with classifications approaches commonly used for LoD in video games and interactive environments [51]. The mean angle value and the range of the last quartile were segmented to obtain three possible scores for geometric fidelity.

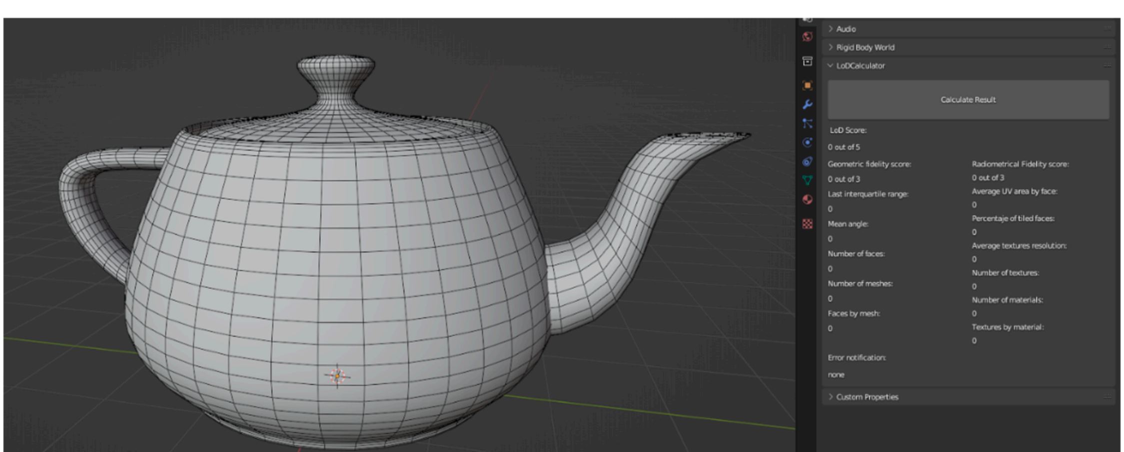

Fig. 1. Interface of LoDCalculator in the right panel. It shows the activation button and the results of the Utah Teapod analysis.

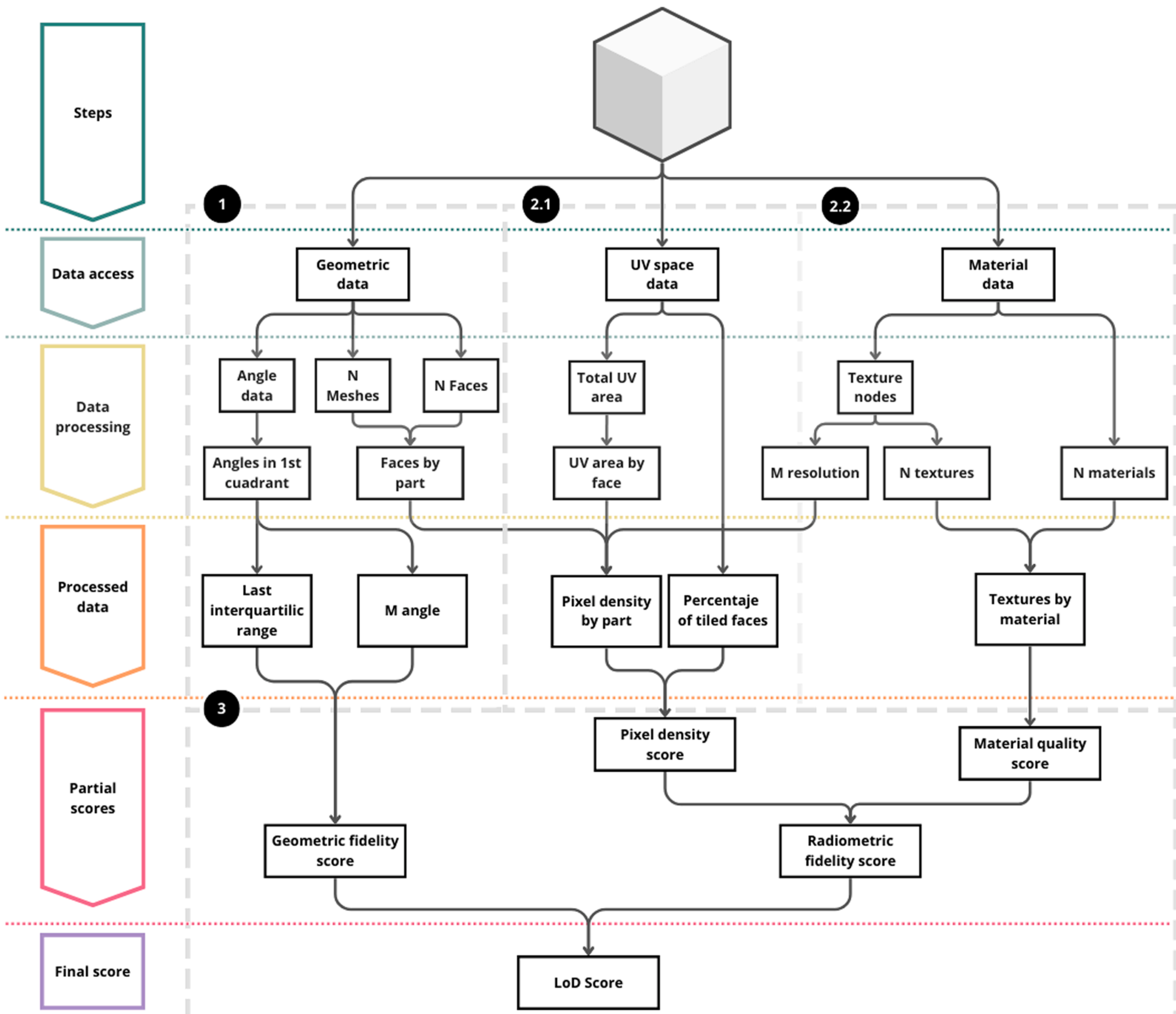

Fig. 2. Summary of the procedure of LoDCalculator. Vertically it shows the steps from the access to the 3D model to the assignment of the score. The gray dashed rectangles separate the main phases of the process.

Radiometric fidelity was calculated by averaging two sub-scores, each with three levels: one based on the proportion of textures per material, and the other on the mean resolution per mesh and the percentage of tiled faces. Finally, both scores were summed and 1 was then subtracted from the result to compute the LoD score. The result was a score ranging between 1-to-5 at half-point intervals on a scale that was similar to other popular scoring scales, such as the Likert scale, making it familiar to a wide range of users. The entire LoDCalculator workflow, including the value ranges used for the thresholds, can be accessed in the LoDCalculator .py code available in the repository.

3. Illustrative examples

LoDCalculator was tested by evaluating 12 models of CH environments, although it can be used in other contexts as well. The results were then compared with evaluations from two groups: the first group consisted of 41 students from 3D modelling courses at the University of Burgos of the bachelor in Video Game Design, and the second group comprised 24 professionals in 3D modelling and animation. Of the 24 professionals, 11 were juniors, meaning they had less than 3 years of professional experience. On the other hand, 13 were seniors, with expertise exceeding 3 years. Those groups operated similarly to the software, assigning scores of 1-to-3 for geometric fidelity, 1-to-3 with half-point intervals for radiometric fidelity, and the LoD score of 1-to-5 with half-point intervals. Before starting the evaluation, some examples were shown to familiarize the users and to obtain more accurate results [52]. For each 3D model, users viewed six renders: 3 clay and 3 with textures with similar lighting. All renders were only illuminated with ambient light generated by a cubemap and rendered using Blender's real-time Eevee engine with default settings to produce comparable images. This subjective method of evaluation has been used in similar research [26,27,30,31]. The models were either from previous group projects (see [53–55]) or downloaded for free from Sketchfab. All models met the software requirements: polygonal 3D models with quad-based topology and primarily texture-based materials. Fig. 3 shows an image of each model with its identification number and LoD score. The complete result of the analysis of these models with LoDCalculator can be found in Annex I hosted in the repository.

On the one hand, Fig. 4 presents the study results with a scatter plot illustrating the relationship between software scores (X-axis) and evaluator's Mean (M) scores (Y-axis) of each model, separated into the student group (green) and professional group (orange). The data have a slight jitter for better visualization. Each score is surrounded by a circle of purple dots with a radius equal to half the distance between scores,

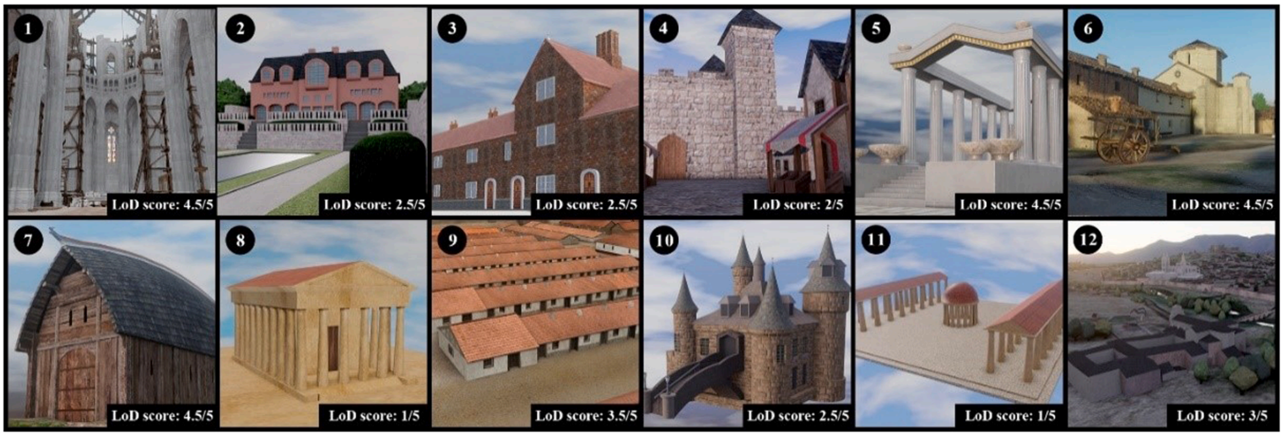

Fig. 3. Images of the 12 3D models analyzed with LoDCalculator, their identification number at the top left, and their LoD score at the bottom right of each image.

indicating the influence area of each score. Data points outside these circles represent outliers. Fig. 4.1 shows geometric fidelity data, Fig. 4.2 shows radiometric fidelity data, and Fig. 4.3 shows the LoD score data. On the other hand, Table 2, Table 3, and 4 present the scores given by the software and the M scores of both groups of evaluators along with their Standard Deviation (SD), with Geometric Fidelity represented in Table 2, Radiometric Fidelity in Table 3, and the LoD Score in Table 4. Finally, Table 5 presents the results of the correlation analysis conducted. Correlation analyses have only been performed on sets of models, as the software score per model does not vary. The ρ have also been extracted from the dataset of both groups. All the data is available in Annex I.

Overall, there was a strong similarity between the sample results and those of the software, except in the extreme scores, where the lack of higher or lower ratings caused the M to diverge. Additionally, except for a few models, the SD were low, indicating a high consistency in the results. In that sense, there were few outliers, with Model 5 having the most divergences followed by Model 7 and 3, though with less noticeable differences. These models are distinguished by higher material complexity, with Models 5 and 7 transferring much of their geometric complexity to normal maps [56], which may have confused the evaluators by mixing geometric and material complexity. Model 3, with its excessively tiled textures, might have also affected perception, especially for less experienced students [11]. It had already been considered that such differences might be harder to assess. Although the software is quite general, greater emphasis and precision were placed on radiometric analysis for this reason. Nonetheless, slight deviations may still arise in more complex models. Outliers that slightly increased the LoD score might be due to the increased complexity of the general analysis of the model and the greater number of options available to the user. Furthermore, the models were large and complex, composed of many parts, which could affect the evaluators' perception. However, the intended purpose of the software was, after all, to analyse complex models and environments. Considering the correlation coefficients, a strong level of agreement was achieved, around 0.750, which is notable for a general-use software like this. The coefficients were slightly lower for students' geometric fidelity, although they were still above 0.670. In all likelihood, it was due to the smaller number of options available to the user in these scores. However, the total ρ in LoD score was almost 0.800, showing a strong relationship. It may also be observed that the professional group was more critical of the models, which could be explained by the novelty effect experienced by the students during the task [57]. In a deeper analysis, it can be observed that junior professionals, despite having a similar perception to seniors, rate the 3D models more positively, indicating that the novelty effect also influences them.

Compared to other experiments validating LoD classification through subjective evaluations [30,51,58], this research features a comparable sample size, with around 50 subjects being common. However, it differs by classifying different 3D models, rather than versions of a single model. Additionally, this research stands out for including subjects with experience in the field, categorized by their expertise. The results are aligned with those of other studies when LoD is evaluated in controlled conditions [58], but in more interactive environments, evaluators tend to have difficulty to differentiate LoD accurately [30,51]. A limitation of this study is the lack of comparative analysis with other classification methods, as the diversity of 3D models and the type of analysis executed are not directly comparable to other measurement methods, typically used for point cloud or distortions analysis [58].

4. Impact

The classification and evaluation of LoD is crucial in 3D model development, as it impacts development time, costs, and enables better management of computational resources on devices, particularly low-resource ones like iVR [33]. However, existing systems have several limitations hindering their widespread use. On one hand, some semantic classification systems lack sufficient guarantees for universal comparisons, are open to broad interpretations, present usability challenges, or have proprietary costs. On the other hand, many studies focus on analysing distortions and complexities of various versions of point clouds or other 3D elements, which are not suitable for video game development, such as iVR [40].

LoDCalculator addresses the need for a universal LoD analysis that is comparable across multiple 3D modelling contexts in an accessible and free manner. This is particularly useful for teams with limited 3D modelling experience due to its user-friendly interface and easy-to-understand results [45]. It is especially valuable in multidisciplinary fields like iVR, where research teams with limited computing expertise often venture into iVR development to broaden their outcomes. Fields such as education and CH dissemination, where iVR is an emerging technology, particularly benefit from this [59]. This research provides a tool that can serve as a common framework for LoD classification, helping to streamline 3D modelling development processes and generating a significant positive impact in these areas, which often lack the resources and experience of industry or engineering [12]. Furthermore, LoDCalculator works through Blender, an open-source 3D modelling suite and one of the most widely used software tools for such purposes [44]. LoDCalculator was tested in that field, yielding good LoD assignment results that were reasonably well aligned with the results of the evaluation group, showing that it is a tool with reasonably accurate results for these purposes.

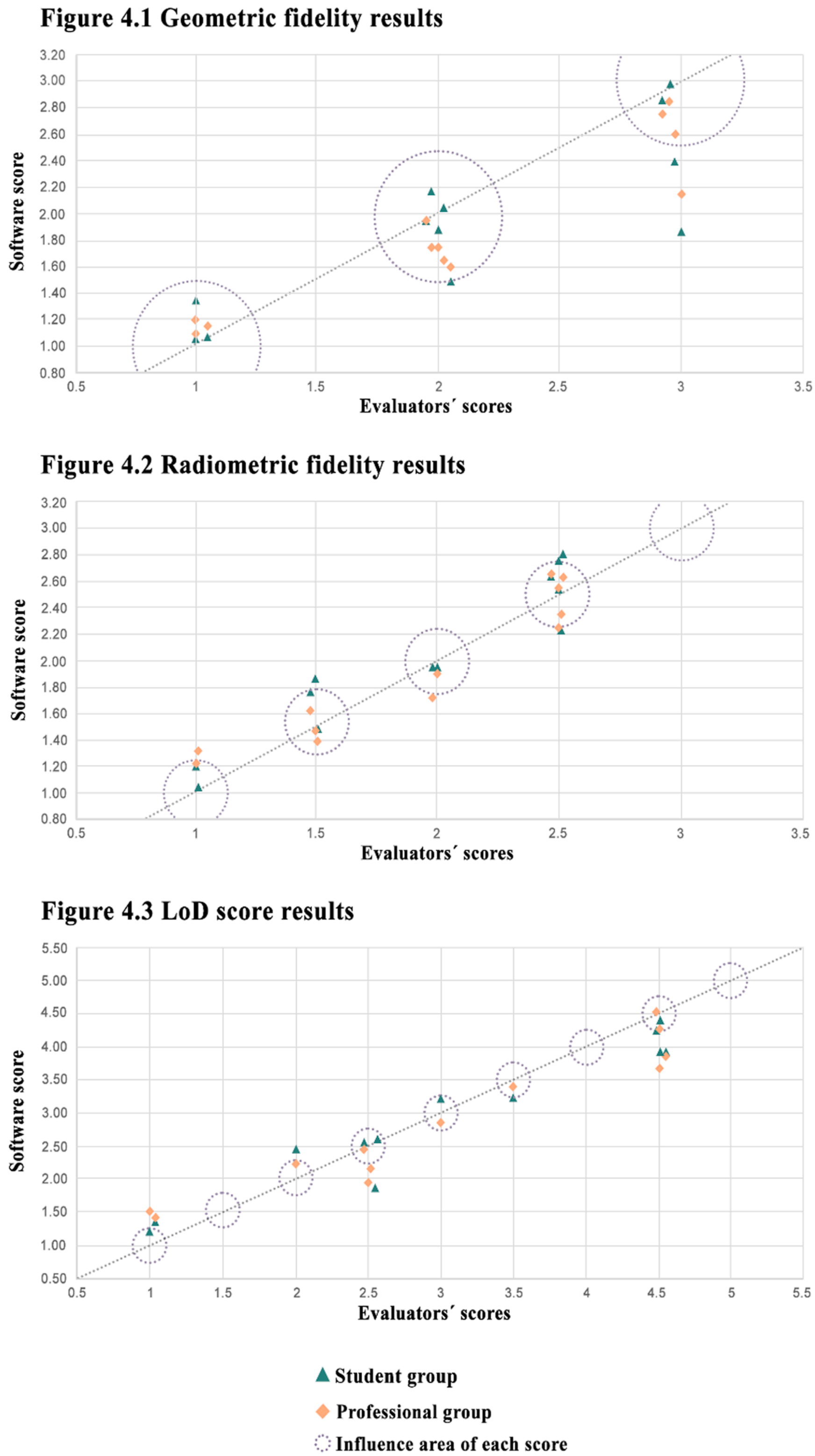

Fig. 4. Evaluation results in scatter plots (M) separated by student group and professional group. Fig. 4.1 shows the geometric fidelity results, Fig. 4.2 shows the radiometric fidelity results, Fig. 4.3 shows the LoD score results.

5. Conclusions

The use of LoD in 3D modelling projects is crucial, although current tools for classifying 3D models by LoD have limitations that hinder their widespread use. The LoDCalculator, an add-on for Blender, developed with the Blender Python API, has been used to address these issues. It features: 1) mathematically defined, unambiguous thresholds; 2) universal analytical capability for the analysis of any model regardless of scale or PC; and 3) system accessibility, integrated into one of the most widely used open-source 3D modelling software with a user-friendly interface. Its main limitation is that it may not be fully suitable for non-realistic 3D models, due to the parameters it applies for generalized

Table 2
Geometric Fidelity results. For each model the LoDCalcualtor score, and the M and SD scores of the Student and Professional group are shown.

Model Number	LoDCalculator score	Student group (M/SD)	Professional group (M/SD)
1	3	2.98/0.16	2.85/0.37
2	2	2.05/0.44	1.65/0.59
3	2	1.49/0.60	1.60/0.50
4	1	1.34/0.53	1.20/0.41
5	3	1.87/0.68	2.15/0.49
6	3	2.85/0.42	2.75/0.44
7	3	2.39/0.59	2.60/0.60
8	1	1.05/0.22	1.10/0.31
9	2	1.95/0.44	1.95/0.60
10	2	2.17/0.54	1.75/0.64
11	1	1.07/0.26	1.15/0.37
12	2	1.88/0.64	1.75/0.55

Table 3
Radiometric Fidelity results. For each model the LoDCalcualtor score, and the M and SD scores of the Student and Professional group are shown.

Model Number	LoDCalculator score	Student group (M/SD)	Professional group (M/SD)
1	2.5	2.63/0.39	2.65/0.33
2	1.5	1.87/0.43	1.48/0.41
3	1.5	1.49/0.41	1.40/0.35
4	2	1.95/0.48	1.73/0.60
5	2.5	2.80/0.31	2.63/0.46
6	2.5	2.76/0.28	2.55/0.46
7	2.5	2.54/0.39	2.25/0.47
8	1	1.05/0.15	1.33/0.29
9	2.5	2.23/0.45	2.35/0.40
10	1.5	1.77/0.39	1.63/0.51
11	1	1.21/0.30	1.23/0.26
12	2	1.95/0.56	1.90/0.55

Table 4
LoD score results. For each model the LoDCalcualtor score, and the M and SD scores of the Student and Professional group are shown.

Model Number	LoDCalculator score	Student group (M/SD)	Professional group (M/SD)
1	4.5	4.24/0.68	4.53/0.50
2	2.5	2.60/0.78	2.15/0.65
3	2.5	1.87/0.75	1.95/0.63
4	2	2.46/0.79	2.23/0.85
5	4.5	3.91/0.86	3.85/0.78
6	4.5	4.41/0.67	4.28/0.73
7	4.5	3.93/0.88	3.68/0.80
8	1	1.22/0.35	1.50/0.51
9	3.5	3.23/0.77	3.40/0.79
10	2.5	2.55/0.67	2.45/0.90
11	1	1.39/0.44	1.43/0.52
12	3	3.21/1.06	2.85/0.95

Table 5
Correlation analysis between the scores of each evaluation group and both groups with the scores of the software in each of its punctuations. For each correlation the Pearson and Spearman coefficients are shown.

Scores	Student group (Pearson ρ/Spearman ρ)	Professional group (Pearson ρ/Spearman ρ)	Both groups (Pearson ρ/Spearman ρ)
Geometric fidelity	0.671/0.669	0.727/0.729	0.689/0.688
Radiometric fidelity	0.785/0.787	0.732/0.749	0.766/0.773
LoD Score	0.795/0.788	0.778/0.790	0.792/0.788

analysis. However, its thresholds can be easily modified for customized use. Comparing the software's criteria with professional and semi-professional users in the analysis of 12 3D models showed a comparable performance to the sample criteria. It is therefore a valuable tool for assessing the LoD of 3D models, especially in multidisciplinary environments, due to its ease of use and integration in the modelling software.

CRediT authorship contribution statement

Bruno Rodriguez-Garcia: Writing – original draft, Software, Conceptualization. **Ines Miguel-Alonso:** Writing – review & editing, Validation, Investigation. **Henar Guillen-Sanz:** Writing – review & editing, Validation, Software, Investigation, Conceptualization. **Andres Bustillo:** Writing – review & editing, Supervision, Project administration, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

Special thanks to Mr. Jose M. Ramírez Sanz from the University of Burgos for his kind-spirited and useful advice in Python language. This work was supported by the Ministry of Science and Innovation of Spain (MCIN/AEI/10.13039/501100011033) and by the European Union NextGenerationEU/PRTR through REMAR Project (CPP2022-009724).

Supplementary materials

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.softx.2025.102107.

Data availability

LoDCalculator Dataset (Original data)

References

[1] Clark JH. Hierarchical geometric models for visible surface algorithms. Commun ACM 1976;19:547–54. https://doi.org/10.1145/360349.360354.
[2] Banfi F. The evolution of interactivity, immersion and interoperability in HBIM: digital model uses, VR and AR for built cultural heritage. ISPRS Int J Geoinf 2021; 10:685. https://doi.org/10.3390/ijgi10100685.
[3] Biljecki F, Ledoux H, Stoter J, Zhao J. Formalisation of the level of detail in 3D city modelling. Comput Environ Urban Syst 2014;48:1–15. https://doi.org/10.1016/j.compenvurbsys.2014.05.004.
[4] Biljecki I.F. The concept of level of detail in 3D city models PhD research proposal. 2013. https://doi.org/10.1145/288692.288701.
[5] Luebke D, Reddy M, Cohen J D, Varshney A, Watson B, Huebner R. Level of detail for 3D graphics. Morgan Kaufmann Publishers; 2003.
[6] Gonzalez-Morcillo C, Weiss G, Vallejo D, Jimenez-Linares L, Castro-Schez JJ. A multiagent architecture for 3D rendering optimization. Appl Artif Intell 2010;24: 313–49. https://doi.org/10.1080/08839511003715212.
[7] Wu S, Hou L, Zhang G. Integrated application of BIM and eXtended reality technology: a review, classification and outlook. Lecture Notes Civil Eng 2021: 1227–36. https://doi.org/10.1007/978-3-030-51295-8_86.
[8] Wesolowski L. Virtual reality and BIM as a potential tool forarchitectural engineers' education. World Trans Eng Technol Educ 2019;17:477–82.
[9] Banfi F. BIM orientation: grades of generation and information for different type of analysis and management process. Int Arch Photogramm Remote Sens Spat Inf Sci 2017:57–64. https://doi.org/10.5194/isprs-archives-XLII-2-W5-57-2017. XLII-2/W5.
[10] Anthes C, García-Hernández RJ, Wiedemann M, Kranzlmüller D. State of the art of virtual reality technology. In: IEEE aerospace conference proceedings. 2016. IEEE Computer Society; 2016. https://doi.org/10.1109/AERO.2016.7500674.
[11] Miguel-Alonso I, Rodriguez-Garcia B, Checa D, Bustillo A. Countering the novelty effect: a tutorial for immersive virtual reality learning environments. Appl Sci 2023;13. https://doi.org/10.3390/app13010593.

[12] Rodriguez-Garcia B, Guillen-Sanz H, Checa D, Bustillo A. A systematic review of virtual 3D reconstructions of Cultural heritage in immersive virtual reality. Multimed Tools Appl 2024. https://doi.org/10.1007/s11042-024-18700-3.

[13] Xie J, Zhang L, Li J, Wang H, Yang L. Automatic simplification and visualization of 3D urban building models. Int J Appl Earth Observ Geoinform 2012;18:222–31. https://doi.org/10.1016/j.jag.2012.01.014.

[14] Volk R, Stengel J, Schultmann F. Building information modeling (BIM) for existing buildings — literature review and future needs. Autom Constr 2014;38:109–27. https://doi.org/10.1016/j.autcon.2013.10.023.

[15] Ronzino P, Amico N, Felicetti A, Niccolucci F. European standards for the documentation of historic buildings and their relationship with CIDOC-CRM. In: Workshop practical experiences with CIDOC CRM and its extensions (CRMEX 2013); 2013. p. 70–9.

[16] Stoter J, Vosselman G, Goos J, Zlatanova S, Verbree E, Klooster R, et al. Towards a national 3D spatial data infrastructure: case of the Netherlands. Photogramm Fernerkundung Geoinf 2011:405–20. https://doi.org/10.1127/1432-8364/2011/00941.

[17] Noskov A, Doytsher Y. Preparing simplified 3D scenes of multiple LODs of buildings in urban areas based on a raster approach and information theory. Themat Cartogr the Soc 2014:221–36. https://doi.org/10.1007/978-3-319-08180-9_17.

[18] Deng Y, Cheng JCP, Anumba C. Mapping between BIM and 3D GIS in different levels of detail using schema mediation and instance comparison. Autom Constr 2016;67:1–21. https://doi.org/10.1016/j.autcon.2016.03.006.

[19] Fan H, Meng L. A three-step approach of simplifying 3D buildings modeled by CityGML. Int J Geograph Inf Sci 2012;26:1091–107. https://doi.org/10.1080/13658816.2011.625947.

[20] Guercke R, Götzelmann T, Brenner C, Sester M. Aggregation of LoD 1 building models as an optimization problem. ISPRS J Photogramm Remote Sens 2011;66:209–22. https://doi.org/10.1016/j.isprsjprs.2010.10.006.

[21] Felicetti A, Lorenzini M. Metadata and tools for integration and preservation of cultural heritage 3D information. Geoinformatics FCE CTU 2011;6:118–24. https://doi.org/10.14311/gi.6.16.

[22] Biljecki F, Ledoux H, Stoter J. An improved LOD specification for 3D building models. Comput Environ Urban Syst 2016;59:25–37. https://doi.org/10.1016/j.compenvurbsys.2016.04.005.

[23] Stoter J, Ledoux H, Meijers M, Arroyo Ohori K. Integrating scale and space in 3D city models. Int Arch Photogramm Remote Sens Spat Inf Sci 2012:7–10. https://doi.org/10.5194/isprsarchives-XXXVIII-4-C26-7-2012. XXXVIII-4/C26.

[24] Zlatanova S, Isikdag U. A swot analysis on the implementation of building information models within the geospatial environment, urban and regional data management. CRC Press; 2009. https://doi.org/10.1201/9780203869352.ch2.

[25] Brumana R, Condoleo P, Grimoldi A, Banfi F, Landi AG, Previtali M. HR LOD based HBIM to detect influences on geometry and shape by stereotomic construction techniques of brick vaults. Appl Geomat 2018;10:529–43. https://doi.org/10.1007/s12518-018-0209-3.

[26] Pan Yixin, Cheng Irene, Basu A. Quality metric for approximating subjective evaluation of 3-D objects. IEEE Trans Multimedia 2005;7:269–79. https://doi.org/10.1109/TMM.2005.843364.

[27] Nader G, Wang K, Hetroy-Wheeler F, Dupont F. Just noticeable distortion profile for flat-shaded 3D mesh surfaces. IEEE Trans Vis Comput Graph 2016;22:2423–36. https://doi.org/10.1109/TVCG.2015.2507578.

[28] Lavoué GA. Multiscale metric for 3D mesh visual quality assessment. Comput Graph Forum 2011;30:1427–37. https://doi.org/10.1111/j.1467-8659.2011.02017.x.

[29] Guo J, Vidal V, Cheng I, Basu A, Baskurt A, Lavoue G. Subjective and objective visual quality assessment of textured 3D meshes. ACM Trans Appl Percept 2017;14:1–20. https://doi.org/10.1145/2996296.

[30] Nguyen D, Hien TT, Huong TT. A subjective quality evaluation of 3D mesh with dynamic level of detail in virtual reality. 2024.

[31] Nehme Y, Dupont F, Farrugia J-P, Le Callet P, Lavoue G. Visual quality of 3D meshes with diffuse colors in virtual reality: subjective and objective evaluation. IEEE Trans Vis Comput Graph 2021;27:2202–19. https://doi.org/10.1109/TVCG.2020.3036153.

[32] Kumar S, Chhugani J, Kim C, Kim D, Nguyen A, Dubey P, et al. Second Life and the new generation of Virtual worlds. Computer 2008;41:46–53. https://doi.org/10.1109/MC.2008.398.

[33] Grande R, Albusac J, Vallejo D, Glez-Morcillo C, Castro-Schez JJ. Performance evaluation and optimization of 3D models from low-cost 3D scanning technologies for virtual reality and metaverse e-commerce. Appl Sci 2024;14:6037. https://doi.org/10.3390/app14146037.

[34] van Oosterom P, Stoter J. 5D Data modelling: full integration of l 2D/3D space, time and scale dimensions. In: International conference on geographic information science. 6292; 2010. p. 310–24. https://doi.org/10.1007/978-3-642-15300-6_22.

[35] Keleşoğlu MM, Güleçözer D. A study on digital low poly modeling methods as an abstraction tool in design processes. Civil Eng Architect 2021;9:2570–86. https://doi.org/10.13189/cea.2021.091513.

[36] Luebke DP. A developer's survey of polygonal simplification algorithms. IEEE Comput Graph Appl 2001;21:24–35. https://doi.org/10.1109/38.920624.

[37] Münster S, Hegel W, Kröber C. A model classification for digital 3D reconstruction in the context of humanities research. Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics). LNCS, 10025. Springer Verlag; 2016. p. 3–31. https://doi.org/10.1007/978-3-319-47647-6_1.

[38] Chopine A. 3D Art essentials the fundamentals of 3D modeling, texturing, and animation. 1st ed. Oxford: Elsevier; 2011.

[39] Obradović M, Vasiljević I, Đurić I, Kićanović J, Stojaković V, Obradović R. Virtual reality models based on photogrammetric surveys—a case study of the iconostasis of the Serbian Orthodox Cathedral Church of Saint Nicholas in Sremski Karlovci (Serbia). Appl Sci 2020;10:2743. https://doi.org/10.3390/app10082743.

[40] Statham N. Use of photogrammetry in video games: a historical overview. Games Cult 2020;15:289–307. https://doi.org/10.1177/1555412018786415.

[41] Pietroni N, Cignoni P, Otaduy M, Scopigno R. Solid-texture synthesis: a survey. IEEE Comput Graph Appl 2010;30:74–89. https://doi.org/10.1109/MCG.2009.153.

[42] Limberger D, Scheibel W, van Dieken J, Döllner J. Procedural texture patterns for encoding changes in color in 2.5D treemap visualizations. J Vis 2023;26:317–33. https://doi.org/10.1007/s12650-022-00874-3.

[43] Zirek S. Synthesising 3D solid models of natural heterogeneous materials from single sample image, using encoding deep convolutional generative adversarial networks. Syst Soft Comput 2023;5:200051. https://doi.org/10.1016/j.sasc.2023.200051.

[44] Kiourt C., Theodoropoulou H.G., Koutsoudis A., Ioannakis J.A., Pavlidis G., Kalles D. Exploiting cross-reality technologies for cultural heritage dissemination. In: George pavlidis (Athena – research and innovation center in information c and KT, editor. Applying Innovative Technologies in Heritage Science, 2020, p. 85–108. https://doi.org/10.4018/978-1-7998-2871-6.ch005.

[45] Mendoza MAD, De La Hoz, Franco E, Gómez JEG. Technologies for the preservation of cultural heritage—a systematic review of the literature. Sustainability 2023;15:1059. https://doi.org/10.3390/su15021059.

[46] Münster S, Terras M. The visual side of digital humanities: a survey on topics, researchers, and epistemic cultures. Digit Scholarsh Human 2020;35:366–89. https://doi.org/10.1093/LLC/FQZ022.

[47] Zhao T, Jiang J, Guo X. A novel quadratic error metric mesh simplification algorithm for 3d building models based on 'local-vertex' texture features. Int Arch Photogramm Remote Sens Spat Inf Sci 2022:109–15. https://doi.org/10.5194/isprs-archives-XLVIII-3-W2-2022-109-2022. XLVIII-3/W2-2022.

[48] Liu Q, Su H, Duanmu Z, Liu W, Wang Z. Perceptual quality assessment of colored 3D point clouds. IEEE Trans Vis Comput Graph 2023;29:3642–55. https://doi.org/10.1109/TVCG.2022.3167151.

[49] Chang JW, Wang W, Kim MS. Efficient collision detection using a dual OBB-sphere bounding volume hierarchy. CAD Comput Aided Des 2010;42:50–7. https://doi.org/10.1016/j.cad.2009.04.010.

[50] Han YL, Chen YP. Image overlapping technique without model for large LoD terrain based on GPU. Appl Mech Mater 2014;556–562:4792–6. https://doi.org/10.4028/www.scientific.net/AMM.556-562.4792.

[51] Vergari M, Warsinke M, Kojić T, Möller S, Voigt-Antons J-N, Abboud O, et al. The influence of the level of detail and interactivity of 3D elements on UX in XR applications. Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics). LNCS, 14706. Springer Science and Business Media Deutschland GmbH; 2024. p. 290–300. https://doi.org/10.1007/978-3-031-61041-7_19.

[52] International Telecommunication Union. Methodology for the subjective assessment of the quality of television pictures BT Series Broadcasting service. 2012.

[53] Rodriguez-Garcia B, Alaguero M. Immersive Virtual reality in cultural heritage dissemination: a comprehensive application for novice users. XR Salento 2023; 2023:287–301. https://doi.org/10.1007/978-3-031-43404-4_18.

[54] Alaguero M, Checa D. Optimización de recursos en la reconstrucción virtual del patrimonio histórico-artístico: modelado 3D de la ciudad de Burgos en el siglo XV. XXII Congreso Comité Español de Historia del Arte 2018 (C.E.H.A.).

[55] Checa D, Rodriguez-Garcia B, Guillen-Sanz H, Miguel-Alonso IA. Framework for developing multi-user immersive virtual reality learning environments. Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics). LNCS, 14218. Springer Science and Business Media Deutschland GmbH; 2023. p. 89–103. https://doi.org/10.1007/978-3-031-43401-3_6.

[56] Deng W, Ke W, Deng Z, Wang X. Virtual design of woven fabrics based on parametric modeling and physically based rendering. Comput-Aided Des 2024;173:103717. https://doi.org/10.1016/j.cad.2024.103717.

[57] Miguel-Alonso I, Checa D, Guillen-Sanz H, Bustillo A. Evaluation of the novelty effect in immersive Virtual Reality learning experiences. Virtual Real 2024;28:27. https://doi.org/10.1007/s10055-023-00926-5.

[58] Vanhoey K, Sauvage B, Kraemer P, Lavoué G. Visual quality assessment of 3D models: on the influence of light-material interaction. ACM Trans Appl Percept 2018;15:1–18. https://doi.org/10.1145/3129505.

[59] Checa D, Bustillo A. A review of immersive virtual reality serious games to enhance learning and training. Multimed Tools Appl 2020;79:5501–27. https://doi.org/10.1007/s11042-019-08348-9.