ISSN 1828-5961

Mattia Sullini
Is a PhD student affiliated to the Dpt. of Architecture at the University of Bologna. His field of research is the digitisation of museum collections, which he approaches with an empirical and experimental attitude, based on his previous experience as CG illustrator and 3D Modeler for Industrial Product, Architecture and Arhaeology

ACCURACY MEASUREMENT OF AUTOMATIC REMESHING TOOLS

Surface Reconstruction from raw Pointclouds generates polygonal meshes characterised by typical local issues that need to be addressed by a global reorganisation of mesh connectivity during the Remeshing process. While in the applications for which remeshing algorithms have historically been developed, the problem can be addressed without affecting the mesh polycount, for the authoring of 3D assets intended for Real-Time Rendering applications, maintaining a low polycount is mandatory. However, there is a trade-off between the number of polygons and accuracy. Hence, in cases where accuracy is a primary concern, such as with Digital Cultural Heritage Objects, the topology of the mesh that ensures this reduction becomes crucial since, to an even polycount, the dimensional discrepancy between the original and different simplified meshes can vary significantly. Therefore, it is vital to evaluate the characteristic quality obtainable from different remeshing algorithms with simplification, but despite the importance of the matter, there is a lack of metrics to perform such comparative benchmarks. This study aims to address this gap by defining and testing a Shape Complexity Index on real-world case studies, which is applicable uniformly and consistently to geometric shapes of generic complexity. Such index allows for pre-determine a target polycount and subsequently normalising

signed distance computations among different test simplified meshes, posing the basis for both quantitative and qualitative analysis of accuracy achievable by any automatic remeshing tool. Additionally, the proposed workflow is employable as an instrument for planning, monitoring, and reporting a digitisation campaign of objects belonging to the Cultural Heritage.

Keywords:
Remeshing; Benchmark; Accuracy; Analythic; Metrics

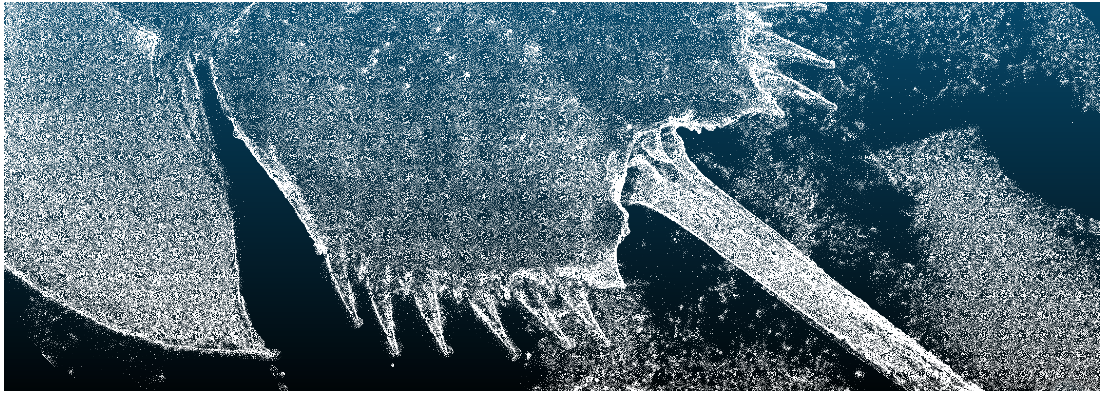
DOI: https://doi.org/10.20365/disegnarecon.32.2024.10

1. INTRODUCTION

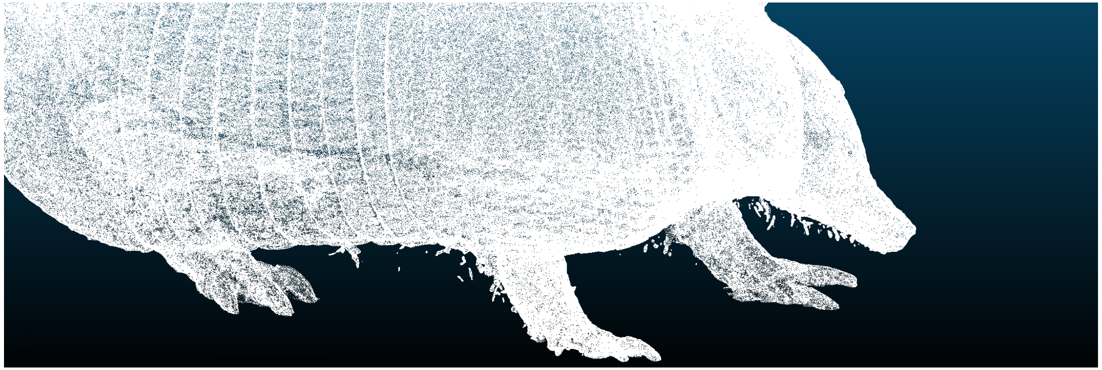
Pointclouds (PC) have been at the heart of the physical object acquisition process since the first contact probes sampled the surfaces of tangible objects by discrete points. Therefore, the whole ecosystem of software, workflows, algorithms, and methodologies related to digitisation has developed accordingly. [Fig.1],[Fig.2]

Such consolidated process is input by PCs as the medium by which measurements of real-world objects are brought into the virtual 3D space, then a discrete polygonal description of measurand's surfaces is reconstructed during the Meshing process by the mean of various algorithms (Sulzer, Marlet, Vallet et al., 2024). The so-generated mesh is typically affected by several local errors, which have to be individuated by a broad range of dedicated indicators during the Mesh Quality Assessment (MQA) step (Sorgente, Biasotti, Manzini et al. 2020) and then corrected in the Remeshing step, which can be approached through very diverse strategies of which Khan, Plopski, Fujimoto et al. (2020) have provided an extensive and detailed literature review.

Most of the methods reported by the authors are based on the reorganisation of mesh connectivity at even polycount because the applications for which the Remeshing algorithms have been developed in the past (like mesh inspection, reverse engineering, or physical simulations) require the densest sampling achievable.



[Fig.1] Raw pointcloud from the Case Study dataset (Limulo). Caption from CloudCompare



[Fig.2] Cleaned pointcloud from Case Study dataset (Armadillo). Caption from CloudCompare

Conversely, there is a direct proportionality between the polycount and the deployment of hardware resources (Wang, Yan, Liu et al. 2018). Hence, a polycount reduction might become mandatory as the computational capacity of the hardware used to process and render the 3D assets decreases from dedicated workstations with locally stored data for professional use down to Real-Time Rendering (RTR) platforms such as Web3D/WebXR or web-based game-engines, which typically both rely on relatively low-performing hardware and need to download the data on runtime.

In such cases, firstly, a Remeshing with simplification is performed to obtain a Low-Polycount (LP) version of the original High-Polycount (HP) reconstructed mesh, then high-frequency features are Baked as 2D Normal Maps onto the LP 3D model (Webster, 2017) [Fig.3]. Indeed, texture maps are processed and rendered by GPUs much faster than actual 3D geometry so that by Baking, it becomes possible to render detailed 3D assets with high Mesh Visual Quality (Abouelaziz, Chetouani, El Hassouni et al., 2020) and minimal impact on hardware resources.

This workflow requires highly optimised meshes [Fig.4] and has been developed within videogame industries' pipelines, but subsequently proved to be suitable also for Digital Cultural Heritage

DOI: https://doi.org/10.20365/disegnarecon.32.2024.10

Objects (DCHO) digitisation workflows, where geometric and colour raw data have to be processed to obtain both a high polycount Master 3D model for documentation purposes, and a low-polycount Derived model for communication and dissemination (Cipriani, Fantini & Bertacchi, 2014).

However, it must be remarked that there is a trade-off between mesh polycount and mesh efficienc, which is a desirable effec, but also with accuracy (Ng, K. W., & Low, Z. W, 2014), which is generally not acceptable apriori for DCHOs, where accuracy is not of secondary importance to apparent Visual Quality. Hence, when processing DCHOs, the polycount reduction must be assessed to a level that makes the digital asset manageable and pliable while simultaneously preserving the most significant degree of accuracy in dependence on the 3D model destination.

Additionally, even if it is generally true that a higher polycount implies a greater accuracy, accuracy can vary significantly even for the same polycount. The reason goes back to the degree of correspondence of the polygonal mesh connectivity with the topology of the digitised geometric form, more specifically with the correspondence between the orientation of local edge loops in the polygon mesh and the principal curvatures of the original surface of the shape, given that curvatures can also apply to discrete geometries (Bobenko, Pottmann, & Wallner, 2010).

Finding the right balance between accuracy and a low polycount is thus a challenging task but is nonetheless mandatory when DCHOs are implied. Additionally, the specific effectiveness of the algorithms used for remeshing complicates it further, so knowing this degree of effectiveness in advance would make it possible to control better the process by which the desired balance is achieved. For reasons that will be explained subsequently, there are currently no objective evaluation criteria for doing so, and this study aims to present one.

2. METHODOLOGY

It is worth highlighting that in the context of surface reconstruction and Remeshing, the term "Accuracy" can be interpreted as consisting of two components, closely interconnected but not perfectly superimposable: Geometric accuracy and Dimensional accuracy. The latter pertains to the extent to which the mesh preserves the geometrical features of the physical object at different scales. Yet, it is not the subject of the present discussion, which is instead focused on Dimensional accuracy, i.e. the discrepancy between a reference entity for the original shape and the simplified mesh [Fig.5]. By measuring such discrepancy, the accuracy of Remeshing performed by a generic algorithm can be assessed consequently both numerically and qualitatively by visualising the outcomes as false colours in a Scalar Field (SF)

Nonetheless, the evaluation that can be drawn from a test conducted on a single mesh is of specific and relative value, firstly because the measurement does not consider the extent of discrepancies when compared the overall dimensions of the object. While this problem can be solved simply by normalising the distances to a reference length proportional to the overall mesh size, an evaluation based solely on the calculation

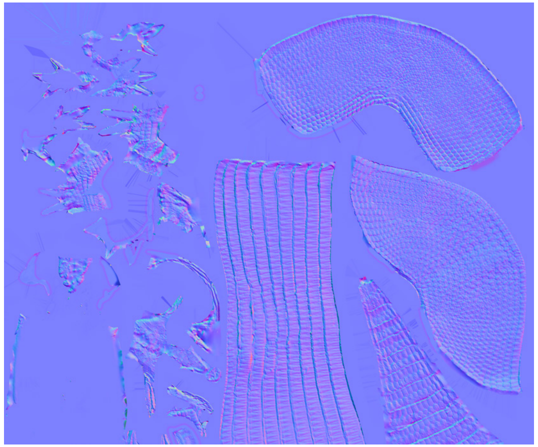
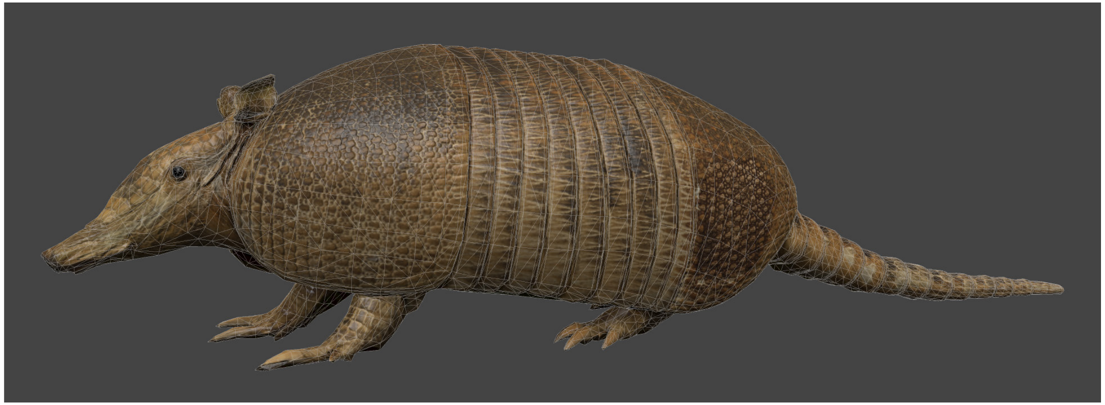


Fig. 3 MikkTS Normal Map obtained from Baking of HighPoly model of Armadillo asset onto a manually retopologised LowPoly version.

Fig. 4.Fully baked LowPoly model of Armadillo from the Case Study dataset. Caption from Modo, a 3D authoring software.



DOI: https://doi.org/10.20365/disegnarecon.32.2024.10

of distances is valid just for that specific mesh and that specific polycount.
To clarify the matter, consider two generic meshes of different geometric complexity having an identical polycount. The shape description would be more accurate in the first case than in the second, and the divergence would increase further the more significant is the difference in complexity between the two shapes. Nonetheless, it is theoretically possible to determine for each shape a specific target polycount which would describe the two shapes with an even level of accuracy. A direct comparison between the two would then be possible, despite their intrinsic difference in geometric complexity. In absence of such objective method, it is equally impossible to compare different algorithms with each other, if not inductively by proofing their behaviour on many different meshes.
Although the concept of complexity is intuitive, in the absence of an objective -hence numerical-definition of it, the operation becomes only possible if it is based on the complexity perceived by a human evaluator, making it suitable for Deep Learning based methods, such as the one proposed by Abouelaziz et al. (2020).
Identifying a complexity index would also allow to preliminarily determine the achievable level of accuracy of a digital asset by comparison with predefined benchmarks, given a generic starting mesh and a target polycount, or to set the level of accuracy preliminarily and subsequently derive the target polycount. Such kind of index, moreover simple to define and obtainable by simple calculations for any given shape, constitutes the core of the present work.
In the following SECTION 3 the proposed complexity index is embedded within a workflow where, starting from a generic mesh M_x, a parameter R_x is generated on the base of the shape coefficient of M_x normalised to a sphere with a shape coefficient of 1, thus making it homogeneously usable for any mesh M_x. This parameter R_x is then used to determine the target polycount of the simplified mesh M_{xS} by basing it on the accuracy obtainable by applying a benchmark polycount to a sphere. On the simplified mesh M_{xS}, distances are measured with respect to M_x, which are then normalised to a measurement proportional to the Bounding box of M_x.
The method guarantees the homogeneity of measurements, which are computed for meshes of variable complexity, but at the same accuracy level. The outcome is then plotted into graphs and analysed to characterise the dimensional accuracy capability of any remeshing algorithm with simplification and, consequently, the comparison between different algorithms.
In SECTION 4, Instant Meshes is considered as case study to proof the method's validity. The software is applied with three different accuracy levels for three meshes differing in geometric characteristics and complexity. The data obtained allows the analysis of the tool's behaviour at diverse levels of simplification in correlation to geometric features at various scales and different local curvature variations.
In SECTION 5, the conclusions that can be drawn from the case study about the proposed method and the criticalities identified are presented. Finally, in SECTION 6 possible further research developments that could use the proposed method are outlined.

3. WORKFLOW

The workflow proposed [Fig.6] is based on a method that moves from a generic dataset of raw PCs $rawP_x$ obtained by digitising physical objects, regardless of the technique used. Even if the ultimate aim is to evaluate the characteristic capacity in terms of dimensional accuracy for any remeshing algorithm, byproduct results are obtained of independent utility while pursuing this goal, albeit within the operational limitations given by the 2-manifoldness requirement for the mesh:

- R_x is a size-independent shape complexity index
- D_T is a target density obtained by predetermined benchmarks
- P_T is a target polycount P_T for different meshes sharing R_x and D_T

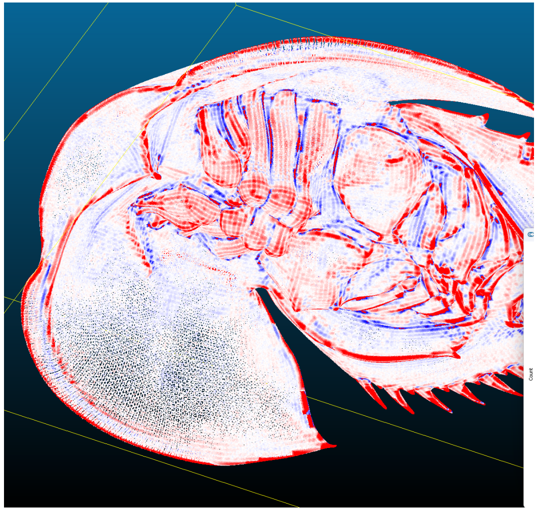


Fig. 5 Signed Euclidean Distance between the original PC and Remeshed mesh of Limulo asset from the Case Study dataset. Caption from Cloud-Compare.

Additionally, it is achieved the possibility of making homogeneous distance measurements between pairs composed by a reference mesh and a remeshed version of it, regardless of meshes' complexity, polycount and dimensions.
The presented workflow is suitable to evaluate any remeshing algorithm or tool in which such an algorithm is employed, but it is optimised for a Tool dedicated to remeshing with simplification in quad-dominant meshes. The presented workflow is optimised for a generic Tool which can be ascribed to this category.
With respect to this choice, it is useful to point out that since the meshes obtained from PCs are usually composed by triangles, remeshing algorithms are designed accordingly. Nevertheless, in the specific case of interest of remeshing with simplification aimed at the preparation of lightweight assets for visualisation in RTR platforms, the available

DOI: https://doi.org/10.20365/disegnarecon.32.2024.10

ISSN 1828-5961

tools typically output quad-dominant meshes as this is the standard for the pipelines built for this type of digital assets preparation (Bommes, Lévy, Pietroni et al.2013).

The workflow includes a preparatory phase, described in subsection 3.1, whose scope is to produce a 2-manifold mesh M_x from the raw starting PC_x with the least loss of significant data possible. Subsection 3.2 describes the process for obtaining the complexity index R_x from the mesh M_x which is then used to determine the target polycount P_T based on the desired polygon density D_T in the manner described by subsection 3.3. The successive subsection 3.4 relates to the Remeshing of the mesh M_x into its simplified version M_{xT} by using any Tool and thus varies according to the specific features of the tool itself. The simplified mesh M_{xT} is then used within subsection 3.5 as a reference for the original mesh M_x to compute the discrepancy between the two in terms of distance.

The distances are stored in a SF which is then exported in subsection 3.6 and fed to a Python script that elaborates and plots them on a linear graph on which a qualitative analysis is conducted to obtain a characterisation of the accuracy achieved for the given starting mesh M_x and the desired polygon density D_T. The graphs are then merged to conduct a comparative inspection both for different densities D_T of the same mesh M_x and for the same densities D_T. across all the meshes. Lastly all are merged for an overall evaluation of the Tool.

Lastly is briefly traced a path for extracting synthetic qualitative indices from the single considered Tool and, consequently, to allow a comparison between several tools in terms of dimensional accuracy which in the present work is qualitatively assessed by the juxtaposition of the distance distribution graphs.

The workflow structure is independent from specific software for the different steps, but in this work is declined for Free Open-Source Software (FOSS). For the individual operations in the attached flowcharts, precise reference is made to the commands available within the software used for the presentation. A detailed guide to using individual commands is not provided as detailed instructions can be found within each software's technical documentation.

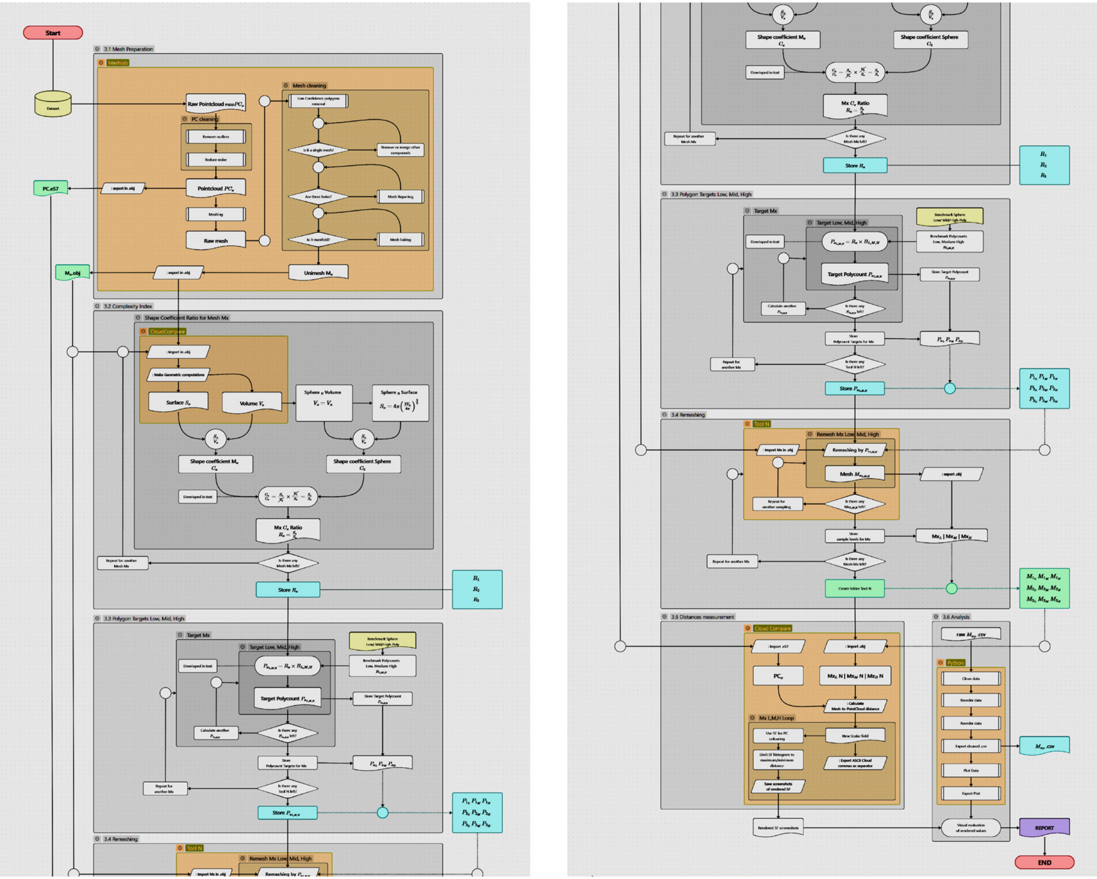


Fig.6 Juxtaposition of initial and final segment of the flowchart resuming the workflow presented in the study. Caption from Obsidian, a knowledge manager with flowchart layouting capabilities.

3.1 MESH PREPARATION

In order to evaluate the accuracy achievable by the Tool at different simplification levels and for different shapes, it is proposed to use 3 different

DOI: https://doi.org/10.20365/disegnarecon.32.2024.10

raw PCs rawP$_{1,2,3}$. The PCs are assumed to have been acquired at state-of-the-art level [Fig.7], with a decent vertex density and the least quantity of lacunae to avoid extensive manual intervention which would jeopardise the objectivity in the assessment.

Even if correctly acquired, such PCs are inevitably affected by systematic errors and false readings that generate noise and outliers (Rakotosaona, La Barbera, Guerrero et al. 2020) that must be cleaned before meshing. For enhanced efficiency and accuracy, it is recommendable to remove outliers before denoising, since this approach allows for a clearer dataset, which in turn enables the denoising algorithms to function more effectively. Different methods are available for both operations, for which Han, Jin, Wang et al. (2017) can be referenced to about denoising and Wang (2014) about outliers' identification and removal.

In CloudCompare both forementioned steps for PC cleaning are grouped within Tools→Clean: SOR filter and Filter Noise. It could be necessary to check the outcome and eventually perform some iterations or manual fencing and removal, especially if the PC has isolated groups of points which might come from false lectures or unwanted elements within the scene.

The outcoming cleaned PC PC$_x$ is exported in .e57 format, chosen over others for being an open format with minimal data loss and generates ductile files even if for the purpose of the present method some part of this data, such as colour, is not used.

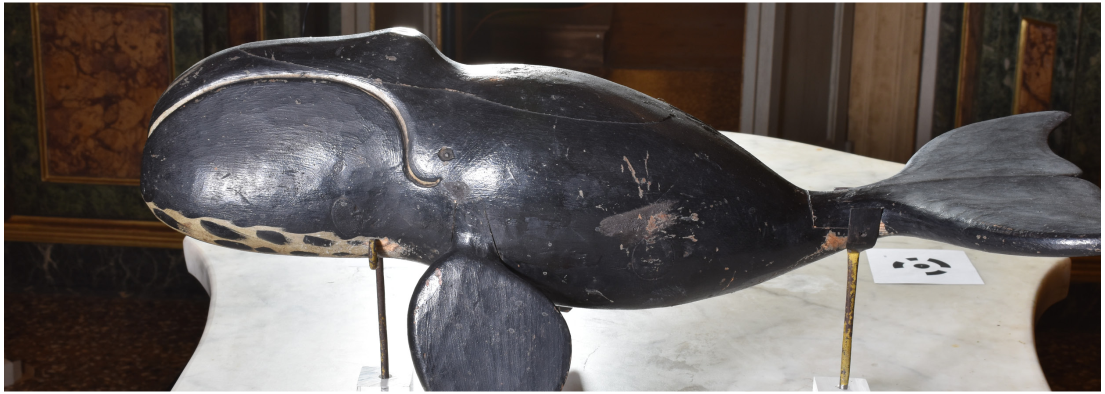


Fig.7 One of the the tangible objects (Balena) belonging to Aldrovandi collection, now at the SMA Bologna, which have been used as a Case Study for the present work.

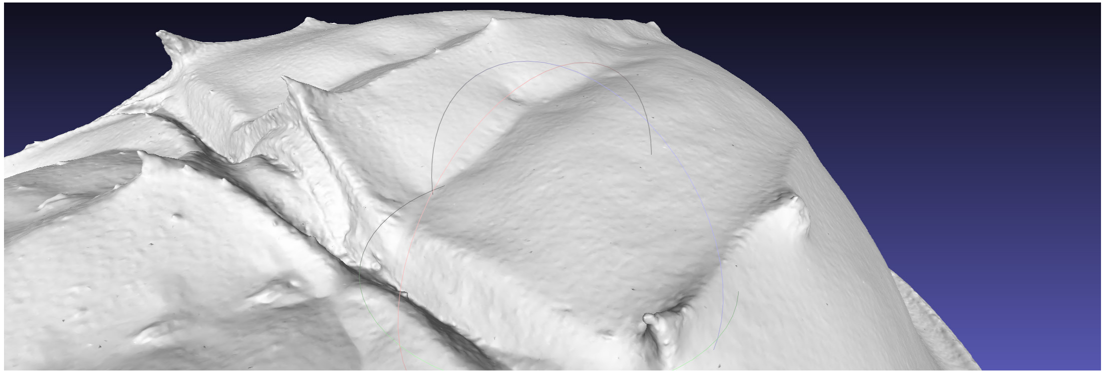


Fig.8 The result of Meshing on Limulo asset PC before the Remeshing phase. Caption from MeshLab.

The PC is then imported in MeshLab for meshing The literature available for this operation is extensive and there are many alternative and well established methods that can be used.

Poisson Reconstruction (Kazhdan, Bolitho & Hoppe 2006) is one of the most diffused algorithms and it is also the algorithm proposed for the present method to be used in MeshLab in its advanced version Surface reconstruction: Screened Poisson that can be found under Filters→Remeshing, Simplification and Reconstruction.

The various software used for digitisation typically include meshing tools, but for the present work is recommended to use MeshLab [Fig.8]. to avoid specific dependencies from each software's toolset and to allow a standardised PC processing phase. CloudCompare has meshing capabilities as well, but MeshLab implements more alternative approaches and more variegated and powerful mesh editing tools.Before meshing, is advisable to compute PC normals since this will ease and make more precise the proper meshing (Tsujibayashi, Inoue & Yoshioka 2018). This can be done within MeshLab by applying the command Compute normals for pointset that can be found under Filters→Normals, Curves and Orientation.

The raw outcome of meshing typically introduces

DOI: https://doi.org/10.20365/disegnarecon.32.2024.10

further errors in the form of poor-quality elements and topological defects such as self-intersecting polygons, as Khan et al. (2020) points out, even if the PC has been cleaned first. The same author presents an extensive list of algorithms that can lead to higher quality meshes, but for the scope of this paper it is suggested to employ the tools offered by MeshLab in the Filters→Cleaning and Repairing submenu by following a progressive refinement which require manual selection using the Filters→Selection submenu. It is required a basic knowledge of mesh cleaning routines which will not be analysed here in-depth for the sake of concision.

As broad outlining of the process, four sub-routines are identified following commonly used identification names: Cleaning, Unimeshing, Repairing and Fairing. Cleaning is addressed to the polygons with a low Confidence level that might have been generated during the Poisson Reconstruction. This step might produce sparse holes in the model that will be closed in a further step. Unimeshing is aimed to identify, evaluate, and eventually remove parts of the mesh that are not connected with the main body of the mesh, which is commonly referred to as Unimesh. Repairing consists in filling mesh holes and to connect eventual separated clusters of polygons that have been evaluated as belonging to the main body of the mesh in the previous sub-routine. At the end of this step the mesh will be composed exclusively by the Unimesh and there will be no holes in it.

The conclusive subroutine of mesh cleaning phase is Healing, which includes all the local interventions on the mesh aimed to achieve 2-manifoldness, amongst which fixing non-manifold edges, removing duplicate vertices, and correcting inverted normals. The outcome of this step is a 2-manifold Unimesh M_x which still holds the most part of the significant geometric data contained within the original PC rawP_x

Once the mesh file is composed exclusively by a 2-manifold Unimesh it is possible to Compute Geometric Computations under Filters→Quality Measures and Computations, which cannot be performed if the mesh has not been prepared earlier as described. It is technically possible to have more than one 2-manifold mesh, but the computations would in this case merge the contributions brought by both meshes since the values needed for the further steps are Mesh Volume V_x and Mesh Surface S_x

The meshes $M_{1,2,3}$ are eventually exported in .obj, which is a widely diffused interchange format that is commonly chosen for its solidity and file dimension when only pure geometry is needed.

3.2 COMPLEXITY INDEX Rx

As it has been previously pointed out, a fixed polycount describes with uneven accuracy two surfaces having different geometric complexity. Consequently, to evaluate comparatively the two remeshing outputs it is necessary to preliminarily determine an adequate target polycount for each, so that the simplification achieved would be equal. In other words, the polycount alone is insufficient to assure that the two surfaces would be simplified to an even level, as it is strictly connected with the geometric complexity of the geometric shape M_x

It is hence mandatory to describe the shape complexity by using and objective index, which could be subsequently used to determine the target polycount related to the desired level of simplification. The shape coefficient C is a good starting point since it weights the surface area S of an object to its volume V as:

$$C = S/V$$

Since S is a value proportional to the square of a length, while V is proportional to the cube, C is not a pure number and is itself expressed as the inverse of a length as:

$$C = (K_1 x L^2)/(k_2 x L^3) = (k_1/k_2)/L$$

Therefore, C varies with the shape's size, even if the geometrical characteristics are the same. For a given polygonal mesh M_x, it's possible to define C_x if its surface S_x and volume V_x are known, which, as we have seen, can be computed for a 2-manifold mesh. However, the variation of C_x with dimensions will follow a law that cannot be determined analytically because even if for simple geometries the value of k_1 and k_2 is known, so it is not for more complex ones such as organic shapes. Thus, since M_x can have any dimension, the value of C_x is generally not comparable for different starting forms.

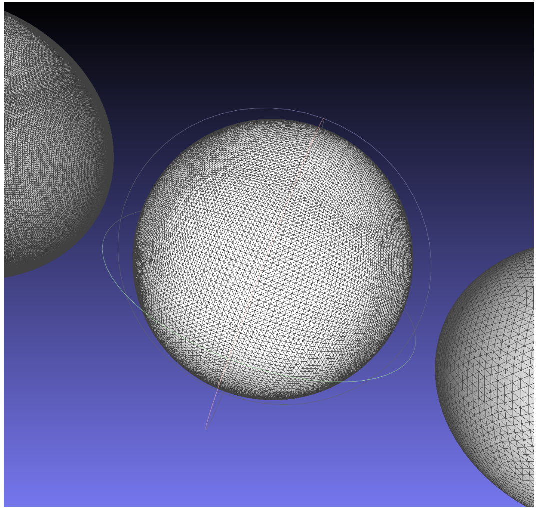


Fig.9 The 3 benchmark spheres whose polygon density is used in the method to determine the target polycounts of any mesh. Caption from MeshLab.

However, it is known that the sphere is the 3D shape for which C_x has the minimum value among all tridimensional shapes of equal volume, as Hermann Schwarz and Henri Poincaré demonstrated in the second half of the 19th century.

Therefore, rather than using the shape coefficient C alone, we can more effectively assess a shape's geometric complexity through the ratio R_x between its C_x and the shape coefficient C_o of a sphere of

DOI: https://doi.org/10.20365/disegnarecon.32.2024.10

equal volume. S_x and V_x can be expressed with respect to volume instead to a length as:

$$S_{xV} = k_{xV}V_{xL}^{2/3}$$

Returning to M_x, its volume V_x is known, hence the surface area of the sphere for which V_x=V will be:

$$S_o = k_o V_x^{2/3}$$

Thus, since:

$$R_x= C_x/C_o = (S_x/V_x)\times(V_x/So) = S_x/S_o$$

Then:

$$R_x = K_x/K_o$$

which shows that the ratio between the volume coefficients of a generic three-dimensional shape and a sphere of equal volume is constant, as expected.
If R_x does not vary, then it also holds when the volume of the generic shape is equal to that of a sphere whose C_o is equal to 1 unit. So, for any mesh M_x, it is always possible to define R_x by few simple calculations as the ratio between its surface S_x and that of a sphere of equal volume S_o. Moreover, since R_x is constant even when M_x is normalised to a sphere with a shape coefficient equal to 1, its value is homogeneous for different geometric shapes, thus allowing them to be evaluated comparatively, as needed.

3.3 TARGET POLYCOUNT SETTING

R_x is hereby the characteristic complexity index of the polygon mesh M_x. If it is assumed that M_x is isotropic, then the density D of its composing polygons will be given by the polycount P divided by the total area of the mesh S. Note that the density in the real cases varies locally, even sensitively, so for the continuation it will be used the average density instead.

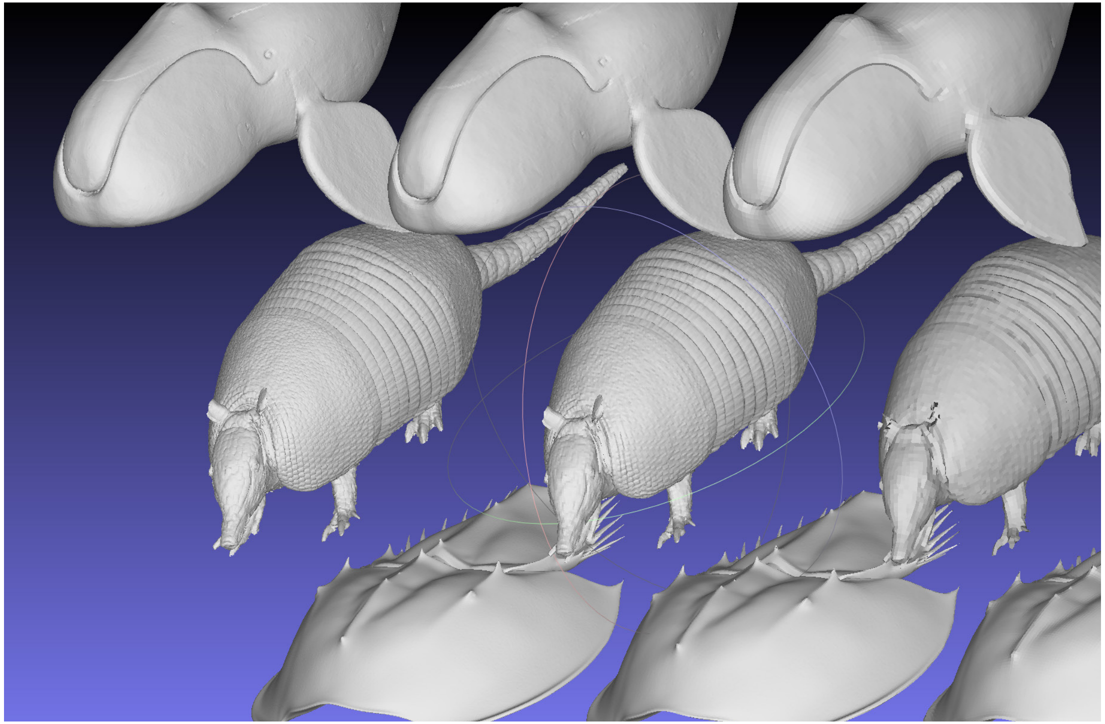


Fig.10 The 3 3D assets (Balena, Armadillo, Limulo) from the Case Study dataset remeshed at the 3 target polycounts and calculated by multiplying the density of benchmark spheres by characteristic Rx. Caption from MeshLab.

$$D_m=P/S$$

The surface of a sphere can be discretised by a triangular isotropic mesh whose actual density D effectively equals the average density D_m. Since curvature is constant on a sphere, the local accuracy with which the isotropic mesh describes the surface is constant making of it an optimal benchmark shape to evaluate the desired density: Since the polygon density is invariant with respect to the dimensions of the surface, as it is a property of the latter, this relationship also holds for the surface sphere S_o that has been used to define R_x therefore for a predetermined mean polygon density D_{om} the unknown target polycount P_{xD} for the mesh M_x which area is S_x will be trivially calculable as follows:

$$P_{xD} = S_x \times D_o$$

It is thus possible for each mesh M_x to determine several target polycounts based on three arbitrary densities that can be evaluated on three benchmark spheres of indifferent size [Fig.9]. In the context of the presented method, 3 densities are proposed:

Low-poly (D_L), Mid-poly (D_M), High-poly (D_H)

DOI: https://doi.org/10.20365/disegnarecon.32.2024.10

which once applied to surface S_x of the mesh M_x yield three polycount targets [Fig.10]:

$$P_{xL} \quad P_{xM} \quad P_{xH}$$

For the definition of each density $D_{L,M,H}$, in the method, it is proposed to separate one from the other by a factor of 4. This is in fact the coefficient of proportionality between a four-sided polygon and its first subdivision by dividing each side in two, being a quad-dominant mesh the optimal output topology for Tool for the reasons that have been recalled above. Therefore, the target polycounts will be proportioned as follows:

$$P_{xL} = 4P_{xM} = 16P_{xH}$$

3.4 REMESHING

M_x .obj is then imported into the Tool, where the 3 target polycounts $P_{xL,M,H}$ are imposed to obtain 3 versions of the original mesh with a differentiated level of simplification: Low-poly, Mid-poly and High-poly. The process is then looped for each $M_{1,2,3}$ and outputs.

$$\begin{matrix} M_{1L} & M_{1M} & M_{1H} \\ M_{2L} & M_{2M} & M_{2H} \\ M_{3L} & M_{3M} & M_{3H} \end{matrix}$$

These simplified meshes are then exported again in format .obj for the forementioned reasons.

3.5 DISTANCE MEASUREMENT

Each simplified mesh M_{xD} .obj is then imported back in CloudCompare along with the previously stored cleaned pointcloud PC_x for distance measurement. The latter is used as the target entity and the former as reference for the command Cloud/Mesh distance, that can be found within *Tools→Distances* which yields for each vertex of PC_x a scalar value for a new Scalar Field (SF) which is mapped with a Blue-White-Red colour scale.

The SF has then been normalised and limited to reference length proportional to the maximum Bounding Box size to make the histograms comparable and the distances homogeneous. In all histograms, the limits $+\%D_x$ and $-\%D_x$ have been set to 1/25000 of the maximum size B_x of the Bounding Box, i.e. 40μ for a 1m Bounding Box. The CloudCompare file is then saved and stored for possible later inspection.

Lastly, SF is exported along with the whole PC as ASCII cloud in .csv format, imposing commas as separator values for easing the subsequent tabulation of data on import. The whole operation is then repeated for each M_{xD} .obj

3.6 DATA ANALYSIS

The M_{xD}.csv file is fed to a simple Python script that

› Removes non-needed values such as point IDs, geometric coordinates, and eventual SF other than the one containing distance computations from the dataset.

›

› Reorders the data records based on Distances' ascending order.

› Saves the cleaned and reordered M_{xD}.csv for eventual later use.

› Normalises the distance values to $\%D_x$.

› Plots the distance series on a linear graph mapping $\%D_x$ on the X-axis that hence ranges from $-\%D_x$ to $+\%D_x$, and on the Y-axis, the single re-

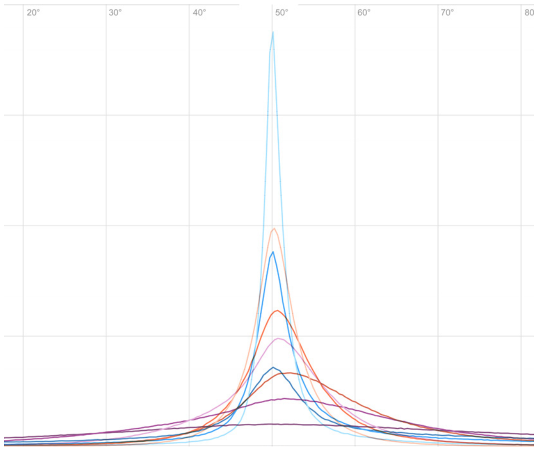
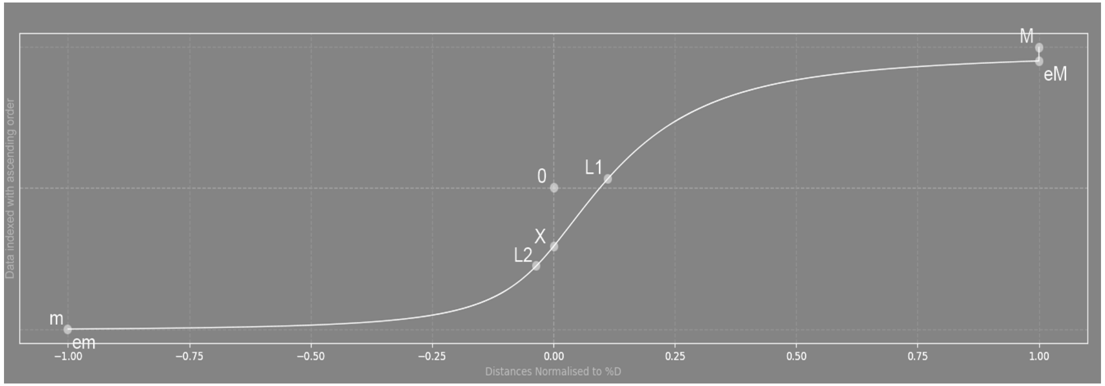


Fig. 11 Statystical distribution of signed distances from 3 remeshed versions of the dataset and the respective cleaned PCs. On X discrete distances intervals, on Y number of occurrencies within the interval. Plot in Python.

Fig. 12 Characterisation and individuation of possible notable discontinuity points along the sigmoid-like graphicisation of signed distances. On X the the distances normalised to 1/25.000 of bounding box maximum dimension, on Y data index by ascending order, normalised to fixed max Y. Python plot.



DOI: https://doi.org/10.20365/disegnarecon.32.2024.10

cords data-index in ascending order

Data are not analysed numerically because rather than a standard statistic evaluation [Fig.11] that would be more suitable to profile overall accuracy, a more qualitative approach would be needed that could identify specific characteristics on the whole distribution of records, especially where accuracy is lower to understand the correspondence between these local "failures" of the Tool when compared to its local geometry and features.
Such an objective would beforehand require numerical indexes whose identification would need to be addressed separately in a dedicated Exploratory Data Analysis (EDA), and this exceeds the scope of the present work.
Nonetheless, this paper provides a brief and basic characterisation of the asymmetrical, sigmoid-like graph [Fig.12], to disambiguate the synthetic qualitative evaluation of the outcomes.

› e_m is the -%D_x abscissa point of the graph. There might be measured distances below this point, but they are ignored in the SF export. The excess is proportional to the length of the vertical segment following the discontinuity in em
› e_M is the +%D_x abscissa point of the graph. The same considerations apply to e_M as to e_m
› 0 is the point the graph would intersect if the distance distribution were symmetrical for positive and negative values, like it would happen if discrepancies were to be ascribable just to instrumental characteristic errors and the graph was consequently a proper sigmoid.
› X is the point where graph shows a (virtual) measurement with perfect accuracy. This point is shifted toward the extreme measurement in the order based on %D_x direction depending on the sign which is recorded the least.
› L- and L+ are the limits of the linear transitional region, where the graph assumes a linear-like trend. The more accurate is the tool, the longer and tilted on the Y direction is the region.
› A- ad A+ are the limits of the plateaus, where errors begin to assume an asymptotic trend. Either L and A points are not strictly defined and are just indicated to highlight the distinct parts composing the typical graph output from the presented setup.

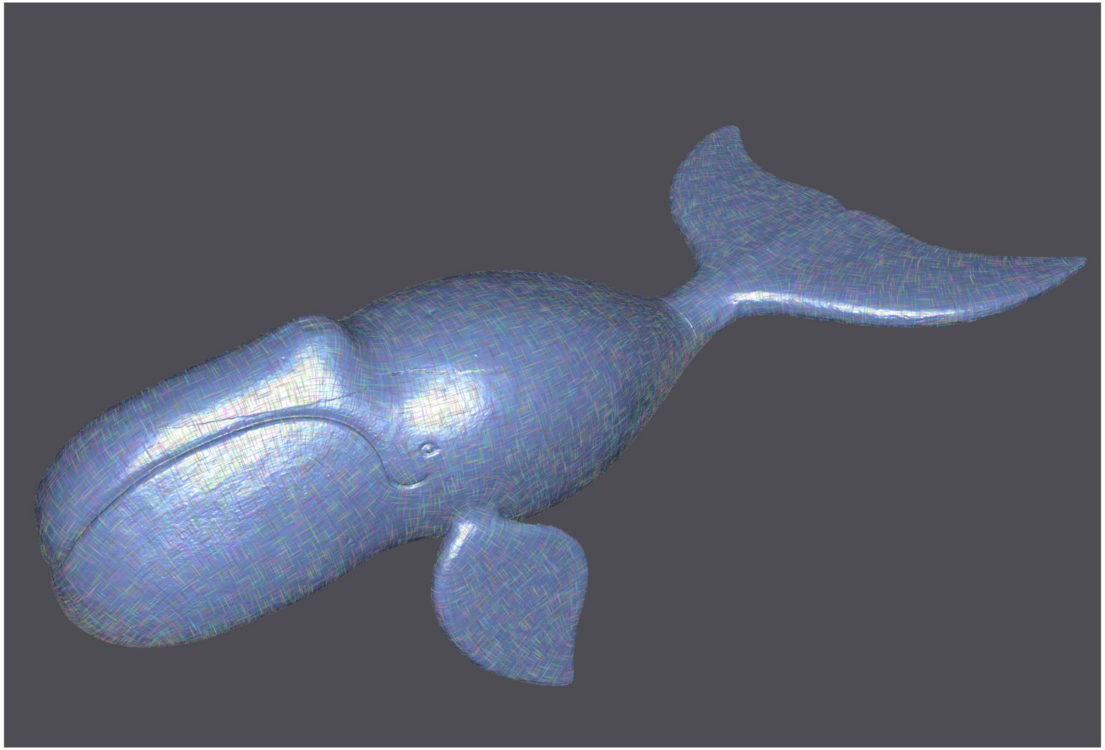


Fig.13 The HighPoly mesh of 3D asset Balena in Instant Meshes. Caption from Instant Meshes.

4. CASE STUDY

4.1 THE TOOL: INSTANT MESHES

Instant mesh is a remeshing tool proposed by Jacob, Tarini, Panozzo et al. (2015) which can output both triangular and quad-dominant meshes, but whose peculiar features are better employed by the latter kind, since before the actual remeshing is it possible to trace several desired directions flows [Fig.13] on the mesh at full resolution, which will be then taken into account for solving the remeshing. Such flows are formed by pseudo-parallel orthogonal edge alignments in a quad-dominant mesh, while they are not univocally definable on a triangular surface since each vertex typically has a valence of 6, consequently making it impossible to set such flows.

4.2 THE DATASET

The following 3 assets have been derived from the dataset built during the digitalisation of the temporary exhibition "L'altro Rinascimento" held

DOI: https://doi.org/10.20365/disegnarecon.32.2024.10

in Bologna from 12/08/2022 until 05/28/2023 (Balzani, R. Barzaghi, S., et al., 2023), coordinated by Cultural Heritage Active innovation for Nex-Gen Sustainable Society (CHANGESs) Extended Partnership [1] for its pilot study aimed at creating the digital twin of the exhibition [2]. The original PCs have been acquired by SfM in purposely sub-optimal conditions; hence, the PC quality is below the state-of-the-art. To improve it, meshing has been performed with a slight downsampling and simplification to remove the most part of local defects. Finally, to recreate the standard conditions on which the method is based, a dense PC has been rebuilt by sampling points on the mesh surface with a Montecarlo algorithm in MeshLab.

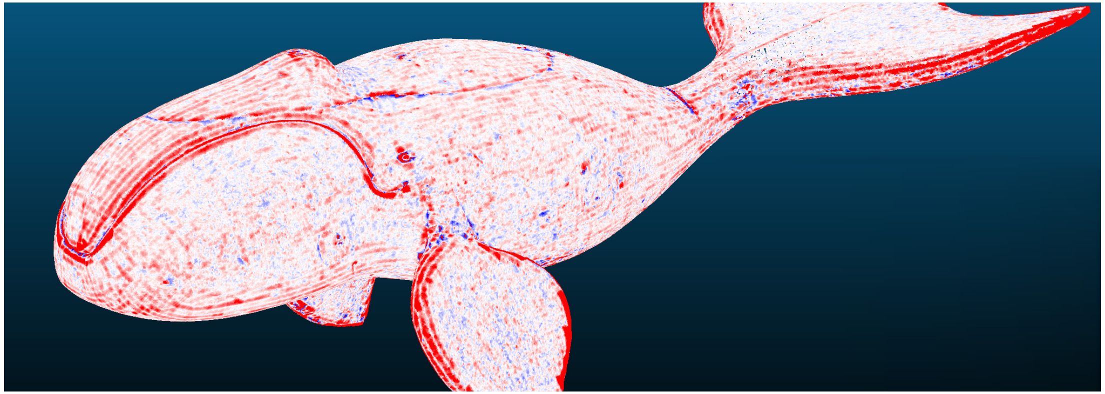


Fig.14 Signed distances (red is positive, blue is negative) between cleaned PC and the Remeshed MidPoly 3D asset Balena.

› Balena is a wooden model of a Whale. It presents a varying almost circular section along the body, with some thin flat appendages for the fins. The surface is overall smooth, with the notable exception of pronounced incisions to describe the mouth line, the eyes, and the blowhole. It has been chosen for the disparity between the overall shape and the scale of the incisions.

› Armadillo is a taxidermied Armadillo. It presents a mixture of features at various frequencies, like visible scutes, ridges, and wrinkles where the animal's skin is thinner. Additionally, it has claws, thin ear flaps, and fur. It has been chosen for its marked medium- and small-scale features.

› Limulo is a taxidermied specimen as well. It presents superficial features at remarkably high frequency and with low depth, so that the prosoma (head) shell can be considered smooth. The border of the opisthosoma (abdomen) presents noticeable thin and long spikes, the telson (tail) is long and thin, with a triangular section. The ventral side is overly complex and articulated. It has been chosen for the challenging geometric features.

›

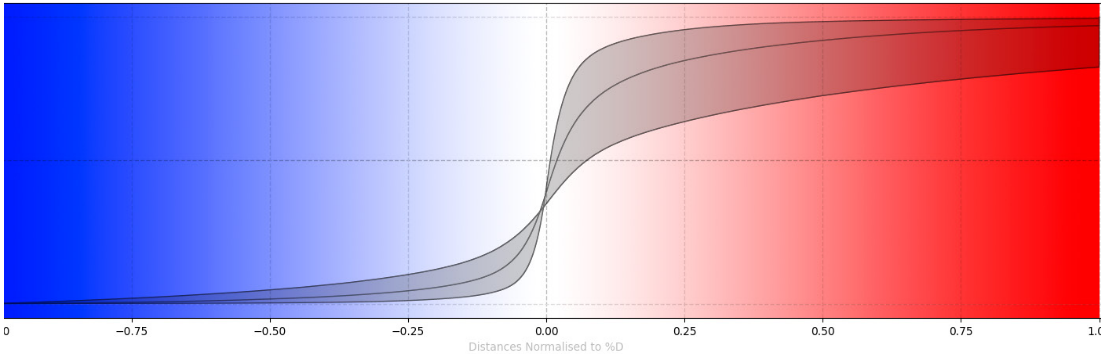


Fig.15 Juxtaposition of normalised Signed Distance graphs for Low, Mid and Highpoly remeshed asset Limulo. Background gradient ranges to full blue for recorded distances even or below minimum normalised Negative Distance, full red for even or above maximum normalised Positive Distance.

4.3 PROCESS

The Case Study has been of significant help to refine the method and the workflow proposed, which thus already implicitly includes the observations done during the development of the Case Study. Consequently, for the sake of concision, the punctual application of workflow in the case study will not be repeated extensively. It's worth pointing out that the screen captions that have been used along the whole work to illustrate it, come from the Case Study itself.

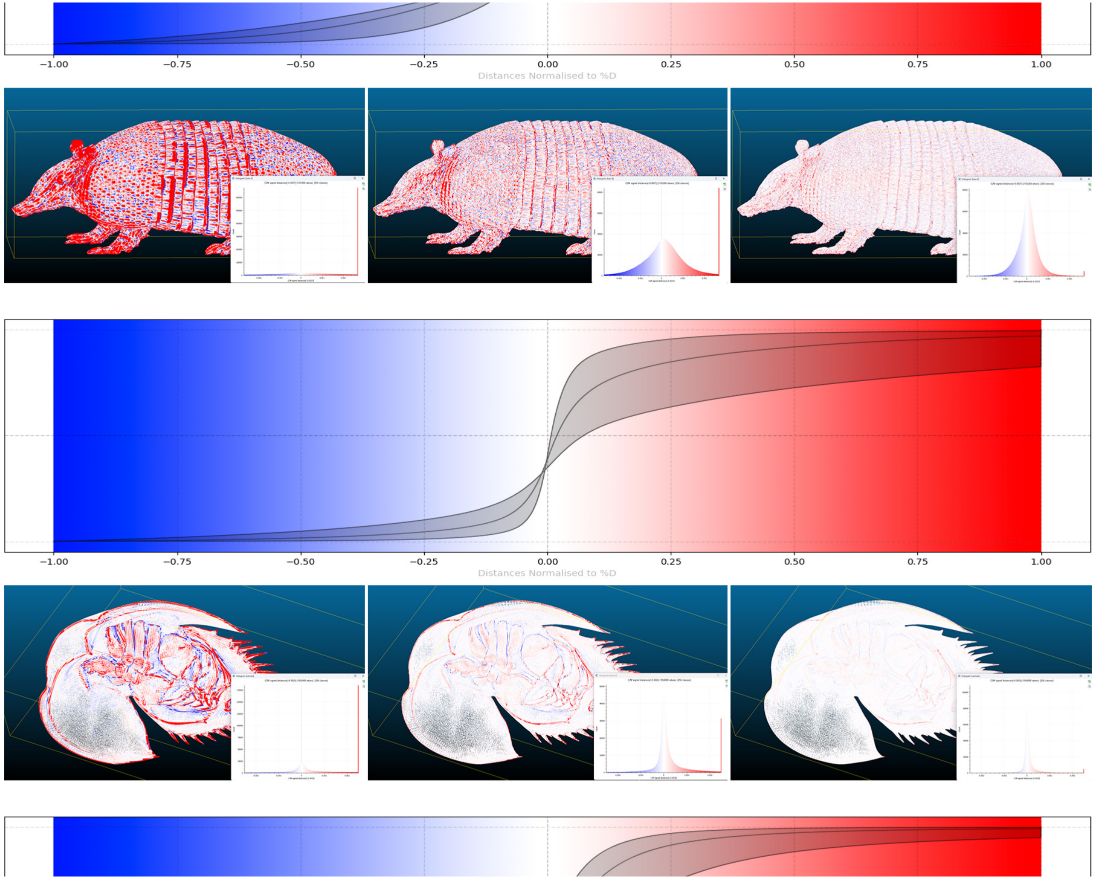
4.4 OUTCOMES PER MESH

› Balena [Fig.14] has a very low dispersion for negative distances, even if when compared with the distribution on the actual PC it is evident that higher errors are associated with the high fre-

DOI: https://doi.org/10.20365/disegnarecon.32.2024.10

quency features. With is respect is noticeable that even for HIGH, e_M is shifted and so there is a certain distance that exceed $+\%D_x$. At LOW, this shift is wider, and the overall precision of positive distances decreases noticeably, firstly with small curvature radiuses and secondarily where the sampling is much higher than the actual shape detail, thus showing the mesh topology. This cannot be properly defined as a lack of accuracy since it is a behaviour connected with the approximation per se un curved surfaces.

- Armadillo is the most problematic case under many aspects. Even at MID some details parts such as ear flaps are missing and the skin folds are irregularly approximated, while at the same time the scutes induce a high uncertainty in the overall shape, presumably because the algorithm adapts the resampled mesh vertices to the closest point of the original mesh not considering whether each vertex belongs to the extruded or recessed part of the scute. The outcome is similar to that of a high instrumental uncertainty and in fact X is relatively close to 0. There is almost no linear transitional region, and the graph is very inclined even in proximity of X. Additionally, many values exceed $+\%D_x$ LOW and probably MID would be unusable without extensive manual editing of the outcome.
- Limulo [Fig.15] lies between the two others with MID that could be used overall. The major defects errors, even if localized, are nonetheless critical and the MID mesh would require interventions especially on the spikes which in some cases are partially missed during remeshing.
-



16. Collage of overall outcomes for the remeshing of the dataset with Instant Meshes.

4.4 TOOL ACCURACY ASSESSMENT

Instant Meshes seems to introduce excessive approximations in correspondence with high frequency features because it outputs isotropic meshes [Fig.16], whose local sampling can result at the same time insufficient for such small features and excessive for more regular parts comporting an unnecessary increment in polycount in comparison with accuracy which could be only addressed by non-isotropic meshes. The tool generally tends to create meshes that are recessed from the base high sampled mesh, thus producing an excess of positive errors rather than negative ones [Fig.17]. Lastly, its general quality is greatly influenced by the preliminary manual setup of mesh flows before the proper remeshing. Despite these outcomes, which can be inferred are shared with any other isotropic quad-based remesher, Instant Meshes is an excellent and documented tool which saves a lot of effort by

DOI: https://doi.org/10.20365/disegnarecon.32.2024.10

providing good quality simplified meshes that can be further edited and improved, especially if the final asset must be low-poly [Fig.18].

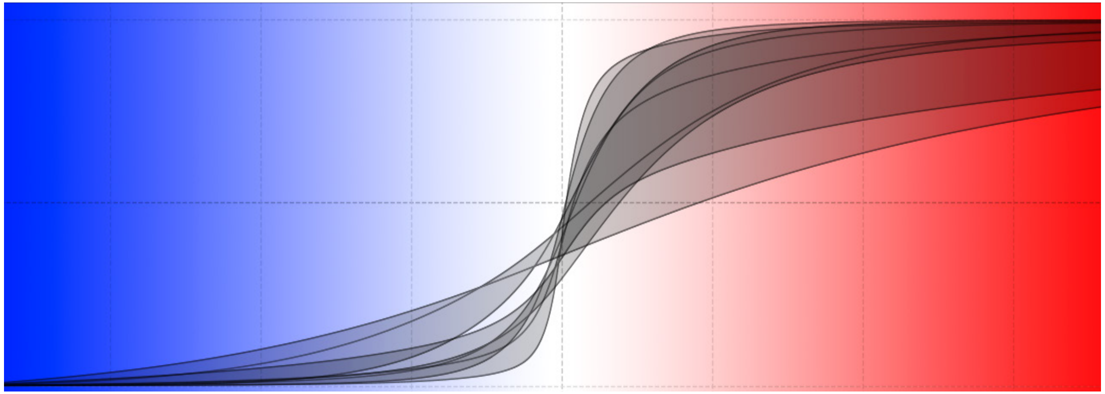
5. CONCLUSIONS

The method proposed introduces numerical and objective indexes that allow direct comparison of different meshes on the base of pre-determined simplification levels. The output measurements are homogeneous and constitute a solid ground for further developments, both qualitative and quantitative.

Even at the present development is already possible to fine tune the average polycount for 3D assets to be employed in RTR applications, even for Web3D/XR environments. Also, the byproducts of the process are of interest, as it has been identified a Complexity Index R_x which is bound to the intrinsic geometric shape of the physical asset and is invariant with dimensions.

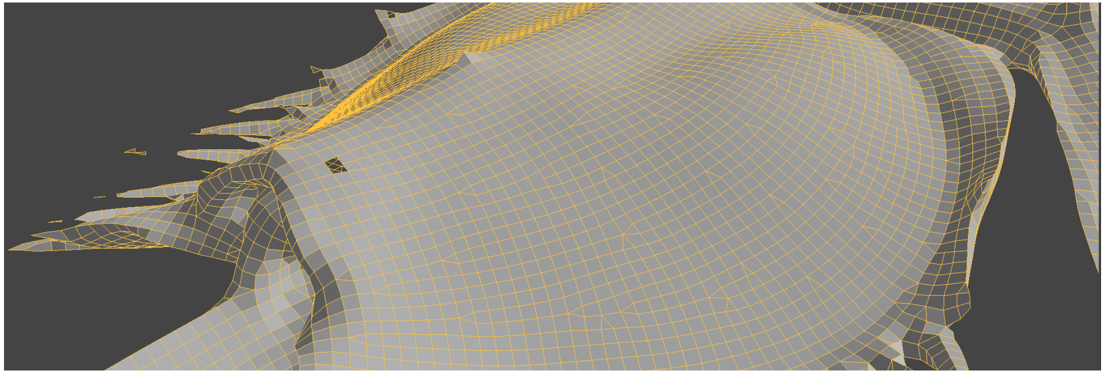
It is worth noticing that the SF associated to measured distances can effectively work as an heatmap which could be used in Deep-Learning based approaches to the remeshing problem in substitution of the pre-training phase, which is typically based on Neural Networks as well, thus easing the process i.e. in Abouelaziz et al. (2020).

The proposed methodology and associated metrics could help addressing the lack of standards, methods, and metrics for the Project Management of digitisation campaigns of DCH, as assessed by the extensive and detailed survey published by the Publications Office of the European Union (2022).



17. Juxtaposition of the normalised Signed Distances graphs for each mesh at each level of semplification with Instant Meshes.

18. Zoom on the Midpoly Remeshed Limulo 3D asset showing topological and connectivity defects.



6. FUTURE WORK

Since mesh complexity varies locally, so should do R_x and all the derived values like polycount etc. R_x is indeed an Average Complexity Index and in a future development it should be developed a method for adapting it locally in order to produce more accurate local target polycounts.

Another weakness of the method is that it requires preliminarily to make the mesh 2-manifold as it is a prerequisite for MeshLab to compute its Volume, but the prerequisite might be achieved also by accepting an approximation and calculating its value on a pre-remeshed and simplified geometry that is easier to edit.

Eventually, the present work proposes a method to assess the distance accuracy of remeshing tools, but nothing is directly assessed about Geometric Accuracy. A solution for this task would need at least to include a comparison with curvature computation and on spatial volume distribution of the shape and is the approach that is under development.

DOI: https://doi.org/10.20365/disegnarecon.32.2024.10

ACKNOWLEDGEMENTS

[1] CHANGESs acts as the Hub within the Hub&Spoke framework, through which all Extended Partnerships are organised in the MiUR actuation of PNRR (https://www.mur.gov.it/it/news/giovedi-07102021/pnrr-le-linee-guida-iniziative-sistema-missione4-componente2).
Specifically, the Spoke 4, led by Università di Bologna, coordinated the cited digitisation campaign's technical aspects. The Spoke 4 granted access to the dataset and allowed its employment according to the illustrated modes.

[2] This work has been partially funded by Project PE 0000020 CHANGES - CUP B53C22003780006, NRP Mission 4 Component 2 Investment 1.3, Funded by the European Union - NextGenerationEU"

REFERENCES

Abouelaziz, I., Chetouani, A., et al., (2020). 3D visual saliency and convolutional neural network for blind mesh quality assessment. *Neural Computing and Applications,* 32, 16589-16603.

Balzani, R., Barzaghi, S., et al., (2023) Saving temporary exhibitions in virtual environments: the Digital Renaissance of Ulisse Aldrovandi. *Digital Applications in Archaeology and Cultural Heritage,* 32, e00309.

Cipriani, L., Fantini, F., et al., (2014). 3D models mapping optimization through an integrated parameterization approach: Cases studies from Ravenna. T*he International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences,* XL–5, 173–180.

Bobenko, A. I., Pottmann, H., et al.,(2010). A curvature theory for discrete surfaces based on mesh parallelity. *Mathematische Annalen*, 348, 1-24.

Bommes, D., Lévy, B., et al., (2013). Quad-mesh generation and processing: A survey. *In Computer graphics forum* (Vol. 32, No. 6, pp. 51-76).

Han, X. F., Jin, J. S., et al., (2017). A review of algorithms for filtering the 3D point cloud. *Signal Processing: Image Communication*, 57, 103-112.

Kazhdan, M., Bolitho, M., et al., (2006). Poisson surface reconstruction. In Proceedings of the fourth *Eurographics symposium on Geometry processing* (Vol. 7, No. 4).

Khan, D., Plopski, A., et al.,(2020). Surface remeshing: A systematic literature review of methods and research directions. *IEEE transactions on visualization and computer graphics*, 28(3), 1680-1713.

Ng, K. W., & Low, Z. W, (2014). Simplification of 3d triangular mesh for level of detail computation. *In 2014 11th international conference on computer graphics, imaging and visualization* (pp. 11-16). IEEE.

Publications Office of the European Union. (2022). *Study on quality in 3D digitisation of tangible cultural heritage: Mapping parameters, formats, standards, benchmarks, methodologies, and guidelines: final study report.*

Rakotosaona, M. J., La Barbera, V., et al., (2020). Pointcleannet: Learning to denoise and remove outliers from dense point clouds. *In Computer graphics forum* (Vol. 39, No. 1, pp. 185-203).

Sorgente, T., Biasotti, S.,et al., (2023). A survey of indicators for mesh quality assessment. *In Computer graphics forum* (Vol. 42, No. 2, pp. 461-483).

Sulzer, R., Marlet, R., et al., (2023). *A survey and benchmark of automatic surface reconstruction from point clouds*. arXiv preprint arXiv:2301.13656.

Tsujibayashi, T., Inoue, K., et al., (2018). Normal estimation of surface in pointcloud data for 3D parts segmentation. *In 2018 IEEE International Conference on Artificial Intelligence in Engineering and Technology (IICAIET)* (pp. 1-5).
Jakob, W., Tarini, M., et al., (2015). Instant field-aligned meshes. *ACM Trans. Graph.*, 34(6), 189-1.

Wang, Y. (2014). *Outlier formation and removal in 3D laser scanned point clouds* (Doctoral dissertation). University of British Columbia).

Wang, Y., Yan, D., et al., (2018). Isotropic surface remeshing without large and small angles. *IEEE transactions on visualization and computer graphics,* 25(7), 2430-2442.

Webster, N. L. (2017). High poly to low poly workflows for real-time rendering. *Journal of visual communication in medicine,* 40(1), 40-47.