

DEFINING COST-EFFECTIVE WORKFLOW FOR A PHOTOREALISTIC 3D CHARACTER BASED ON A REAL PERSON USING A METAHUMAN FRAMEWORK

Igor Fijat, Neda Milić Keresteš , Bojan Banjanin 
University of Novi Sad, Faculty of Technical Sciences,
Department of Graphic Engineering and Design, Novi Sad, Serbia

Abstract: *The average estimated time for creating a realistic human-like 3D game character using traditional workflow is approximately one hundred hours. Creating corresponding facial animations for the persuasive narrative of 3D characters is even more time-consuming. The manuscript explores the possibilities of creating a personalized digital character according to an actual person within the newly-developed technology. The goal of the work involves defining guidelines for the workflow of generating a personalized three-dimensional character based on a real person in the MetaHuman framework. The given detailed insight into the state-of-the-art methodology for creating game characters ensures understanding and reproducibility within a timeframe that is ten times shorter compared to the standard 3D character design workflow.*

Key words: character design, game engine, MetaHuman, Unreal, photogrammetry, 3D modeling

1. INTRODUCTION

Iconic characters are much more than just drawings. In visual art, character design is the complete creation of a character's aesthetics, personality, behaviour, and overall visual representation. He is the bearer of the story, and every aspect of the character, like shape, colour, and detail, is chosen for a reason. Personality traits often suggest a character's appearance, although the reverse is true. Character design aims to create visually attractive, recognizable characters that will easily connect with people. Characters are undoubtedly one of the most important assets of a game, affecting the entire game experience.

The evolutionary course of the appearance of the character can be observed from two aspects - the visual depiction conditioned by the hardware-software display capabilities of the device and the development of trends in character design itself.

In the early days of gaming, what were called game characters were just a few pixels on the screen. With the improvement of the hardware, these characters got more details, whether related to visual representation, animations, or voices. Over time, hardware development has come a long way, allowing display fidelity to unimagined limits. On the other hand, the evolution of character design takes place more slowly, adapting, among other things, to modern social conventions (Iwaniuk, 2017).

The main phases in the traditional workflow of creating a digital 3D character are:

- Profiling - The character must be presented to the player so that he understands the needs and ambitions behind a particular behaviour of the character (Lankoski, 2002). For each character, it is necessary to determine the point of view and attitude since they prompt action and suggest the nature of the character. Similarly, weaknesses or flaws are useful guides for players toward appropriate behaviour. Also, imperfections or customizations make a character more interesting.
- Concept art - The artist explores various sources, finds inspiration and sketches ideas taking into account the profile defined in the previous step. Artists must research references and compare their designs to others to avoid too much resemblance to existing characters. The design must be unique and recognizable. After the moodboard, artists create their brief, where they try out various shapes and figures that they come up with in the form of short drawings and sketches, which help them decide on the best option according to their needs and requirements (Kuntjara & Almanfaluthi, 2017).
- 3D modeling - The evolution of 3D technology has transformed the gaming industry, enabling the creation of unique and realistic content. The 3D modeling phase can generally be divided into

blocking, sculpting, creating an adequate topology - retopology, unwrapping (and baking), and applying textures (Rajpurohit, 2022).

- Rigging and skinning - To make animation more accessible, a system of auxiliary elements is set up, the management of which is reduced to deformations and basic controls of translation, rotation, and scaling. Setting up such a system is called rigging. The elements with which the polygon mesh can be easily and realistically moved are descriptively called the skeleton. Although there are several types of rigging, the one with the skeleton and joints for rigging the whole body is most often used. Apart from it, face rigging is also often used. By rigging, therefore, a skeleton is created for the desired 3D model, which behaves in accordance with predefined principles. Connections between certain parts of the skeleton are defined by the hierarchy and based on it, it can be determined which elements of the skeleton will affect the movement of others in the hierarchy. In order for the skeleton to be able to drive the model, it must eventually be connected to it by defining connections between the skeleton and points on the model, in a process called skinning (Rajpurohit, 2022).
- Animating - Animations of the character's body movements and facial expressions must reflect his personality traits (Rajpurohit, 2022). Movements must look natural and believable to ensure an adequate experience. The created skeleton can greatly facilitate the animation process. The position of each bone (as a part of the skeleton) is defined by animation variable (abbreviated AVARS).

2. A PERSONALIZED 3D CHARACTER CREATION WORKFLOW

The topic of the research is the creation of a digital three-dimensional realistic personalized character modeled on a real person using the Metahuman working framework, as well as the implementation of such a character in Unreal game projects. The MetaHuman uses the information of the 3D network of the cephalic region, mapping the characteristic facial features of a real person, on the basis of which the initial head model is generated as the closest approximation of the input parameters (Unreal Engine, 2022).

The stages of creating a character using a suggested methodology can be divided into five steps:

1. Generating a digital 3D model by photogrammetry,
2. Correcting the topology of the model,
3. Importing the model into Unreal Engine 5 and applying the MetaHuman plugin,
4. Completing the MetaHuman model in the MetaHuman Creator application,
5. Implementing the character in the Unreal Engine 5 project.

2.1 Generating a digital 3D model by photogrammetry

Taking into account objective factors when making a decision, such as simplicity, speed, and practicality of production, reversible engineering will be used for the given task, i.e., creating a model based on an already existing physical object using a method called photogrammetry, with the help of an android mobile device and a free version of the application Polycam.

After photographing the desired subject at least 20 times from different angles, the user is given an overview of the series of shots with the possibility of removing or adding photos to the series. From the additional options, the user can choose the desired level of detail (optimized/medium/full/raw) depending on the needs and later application of the model, as well as object masking to help separate the subject from the background. After settings, the photos can be uploaded and processed, obtaining a finished polygon network ready for export (Figure 1).



Figure 1: Polycam model

2.2 Correcting the topology of the model

The models created in the previous step under the described conditions cannot be directly imported into the Unreal Engine. When reproducing the topology in the previous step, specific errors may occur, such as gaps, i.e., missing polygons; degenerate polygons; overlapping polygons; incorrectly oriented normals; the appearance of a non-manifold topology. Although the MetaHuman plugin can tolerate these errors to a certain extent, specific overlapping errors and non-manifold topology should be removed before importing into Unreal Engine.

In addition, when processing the captured photos, the algorithm takes into account a more extensive area compared to the region of interest. In other words, the initial model contains unnecessary parts that can cause problems in further steps, which is why it is advisable to "purify" the model, leaving primarily the region that will be analysed when creating the MetaHuman model. And finally, using the MetaHuman plugin requires importing files exclusively in FBX or OBJ format. Therefore, before importing the model into Unreal Engine, it should be prepared for further steps in some of the suitable software tools, such as Autodesk Maya, 3ds Max, Meshlab, Blender, etc (Figure 2).

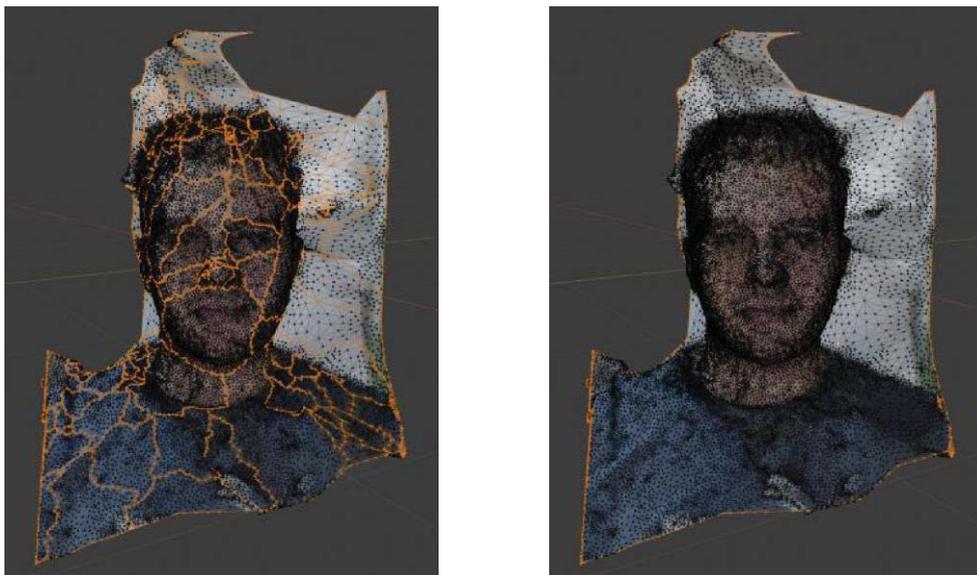


Figure 2: A 3D model in Blender software before and after correcting non-manifold topology

2.3 Importing the model into Unreal Engine 5 and using the MetaHuman plugin

Having a ready, geometrically and topologically appropriate model, the following tools will be used in the next step:

- **MetaHuman Creator (MHC)** - MHC is a free, cloud-based tool that allows you to easily and quickly create fully rigged, photorealistic digital models of people in a browser window.
- **Unreal Engine 5 (UE5)** – UE5 installation can be found on the official website provided by Epic Games company. The most critical novelty UE5 compared to the previous UE version is the Mesh to MetaHuman functionality. This plugin allows you to create a fully rigged, ready-to-animate MetaHuman character based on the chosen model (mesh). The starting point is, therefore, a mesh of polygons, i.e., a 3D model of the head with texture data generated in the process of 3D scanning, sculpting, or traditional modeling. The Mesh to MetaHuman function uses automatic landmark tracking on the model to apply a MetaHuman topology template to it, meshing the resulting head with one of the existing predefined models for the rest of the body. This template is then sent to the cloud, where it will be matched with the closest MetaHuman model from the database. After that, the rigged MetaHuman is downloaded or opened by the MHC application and refined and corrected. Certain elements, such as hair and face texture, must be applied subsequently in MHC or another application, which is why it is not necessary to pay too much attention to these regions when forming the input model of the desired person (Unreal Engine, 2022).
- **Quixel Bridge** - This is an application that will fulfil the task of exporting and downloading MetaHuman models. It acts as a link between UE5 and MHC tools but also offers a library of ready-made models that can be used in projects. After importing the model and enabling the MetaHuman plugin to work, it is necessary to create a MetaHuman Identity asset from the MetaHuman submenu, which can be opened in a new window by double-clicking. The main toolbar suggests the order of the operations. Using the Components from Mesh option, the MetaHuman model will be generated based on the chosen, previously imported polygon mesh. The following step is the selection of representative frames in a neutral pose, which is analysed while processing the final model's topology (Figure 3). It is possible to track and analyse other frames besides the neutral pose by selecting them with a Promote Frame option. After promoting at least one frame and activating markers as representatives of the analysis (Promote Frame and Track Active Frame options), the MetaHuman Identity Solve option becomes available. By selecting it, the points of the template model will be adjusted to the volume of the neutral pose mesh that was analysed in the previous step. The resulting model can be viewed in the viewport window and compared with the initial polygon mesh. Finally, after selecting the Body component and the desired body, the created template mesh can be forwarded to the MetaHuman backend by pressing the Mesh to MetaHuman button (Figure 4). The final MetaHuman model can now be accessed with the help of the Quixel Bridge application for download and the MHC application for additional corrections and finishing of the model

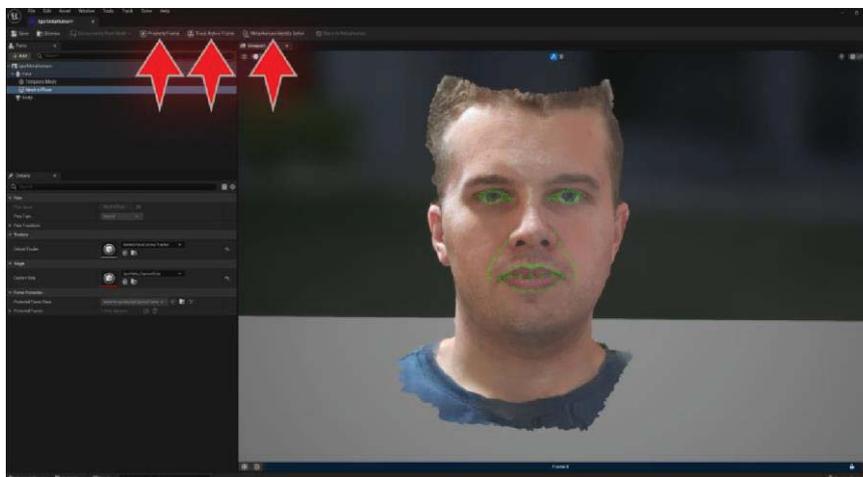


Figure 3: Promote Frame, Track Active Frame and MetaHuman Identity Solve options

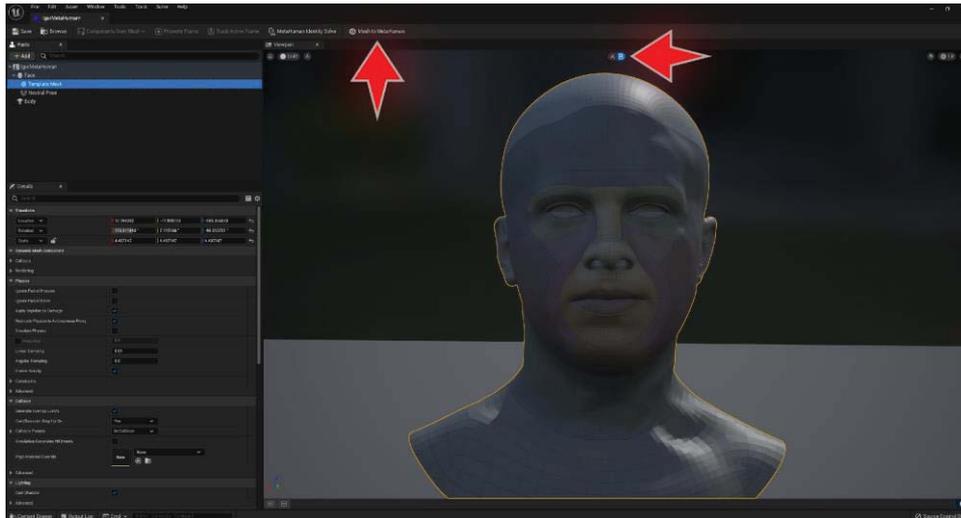


Figure 4: Overview of the final model and exporting to MetaHuman backend (Mesh to MetaHuman option)

2.4 Completing the MetaHuman model in the MetaHuman Creator application

The user interface of the MHC application is self-explanatory and intuitive to use so that every user with a short familiarization and getting used to it can effectively use all the current possibilities: correcting the model topology, applying the skin texture, selecting the characteristics of parts of the face (eyes, teeth, applying makeup) and hairy regions of the head (hair, eyebrows, eyelashes, moustache, beard). Personalizing the rest of the body in the app is limited to selecting some predefined body proportions, clothing, and footwear.

For MetaHuman models created using the Mesh to MetaHuman plugin, there is an additional tab with options called Custom Mesh (Figure 5). As stated earlier, Mesh to MetaHuman works by finding a predefined MetaHuman model that is the most similar result to the mesh created in the previous step. Additionally, it allows changing the mesh of the matched predefined MetaHuman model by tweaking the amount of volume influence for certain parts of the head (blending the predefined model and generated mesh). These differences from the predefined model are desirable since they make the model unique. However, in some instances, the algorithm could take undesirable parts (such as hair or an accessory) and create a topology that does not match the shape and characteristics of a real person. In this case, increasing the proportion of the predefined model's influence is appreciated if it helps to obtain a more accurate geometry.

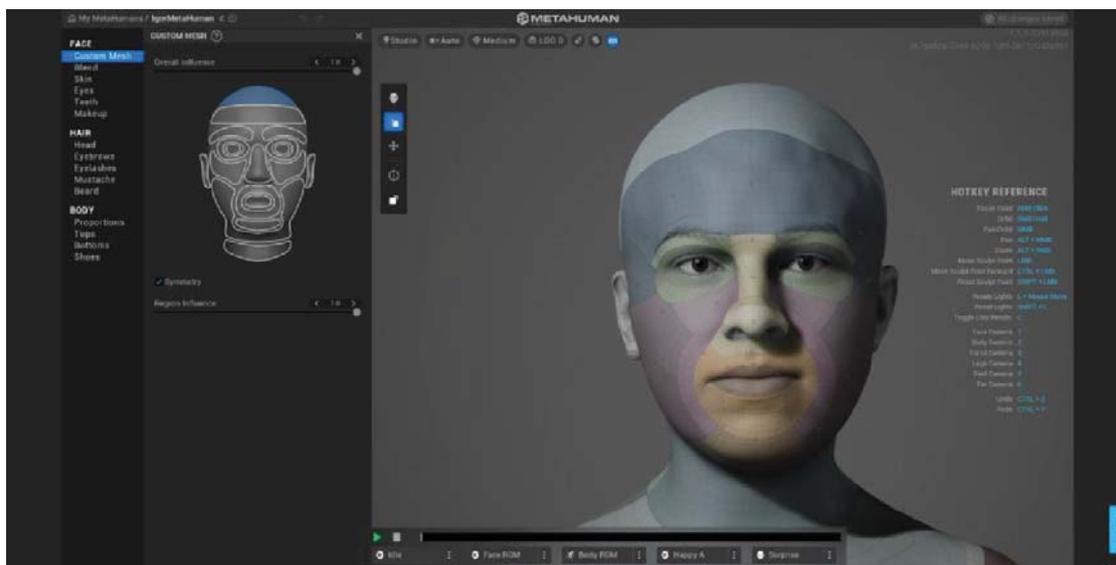


Figure 5: Custom Mesh and markers in the MHC tool

The Sculpting Toolbar contains very useful Blend, Sculpt and Move tools. Blend has the purpose of blending, i.e., mixing the proportions of the features of the selected characters' faces. By selecting the Sculpt tool, markers appear on the face that can be used to reshape the region controlled by the particular marker. The influence of each marker is limited to a certain area, while the zones affected by the markers can overlap. Unlike Sculpt, which controls individual characteristics, the Move tool manages a group of markers at once, allowing faster, more drastic model changes. The final result of the correction of the input model is shown in Figure 6, while its comparison to the real person is demonstrated in Figure 7.

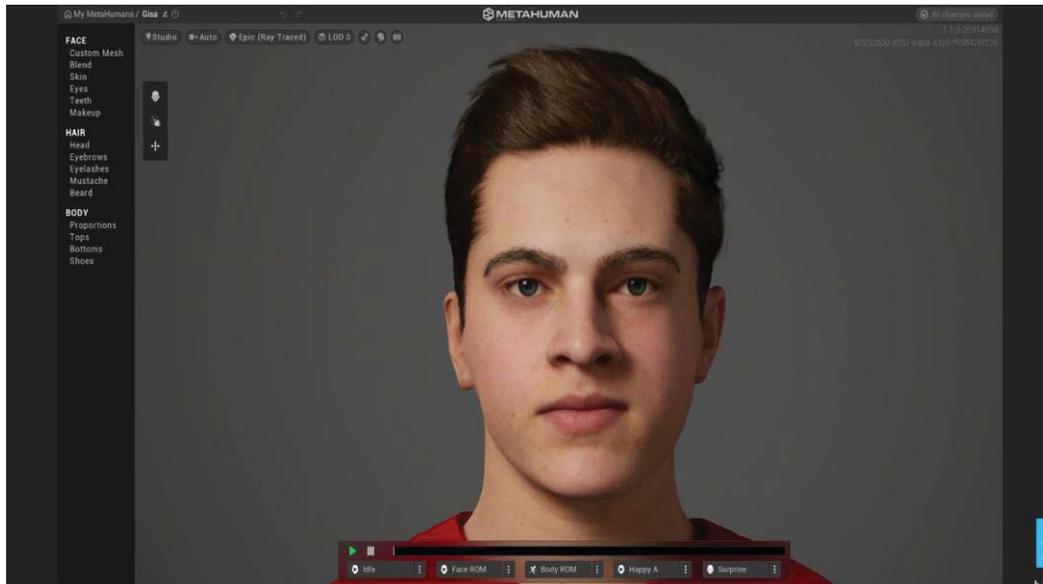


Figure 6: The final result of MetaHuman model correction



Figure 7: Comparison of the real person and the created 3D character

2.5 Implementing the character in the Unreal scene

The implementation of the MetaHuman character into the Unreal project is done through the Bridge tool by selecting it, downloading it, and then adding it to the open project. The character can be used in any imaginable way in games, movies, or presentations... To demonstrate the completed scene, the resulting character is inserted into the project with an iconic environment downloaded from the Unreal Engine resource store, as seen in Figure 8.



Figure 8: The scene with the MetaHuman character of the UE 5 project

3. CONCLUSIONS

The research dealt with creating a photorealistic three-dimensional model based on a real person in the MetaHuman framework. The complete creation of a personalized 3D game character based on a real person is conveyed in an extremely short period, with almost no need for 3D modeling skills. The application and improvement of this technology lie within the limits of the imaginable.

4. ACKNOWLEDGMENTS

This research (paper) has been supported by the Ministry of Education, Science and Technological Development through project no. 451-03-68/2022-14/200156 “Innovative scientific and artistic research from the FTS (activity) domain”.

5. REFERENCES

Iwaniuk, P. (2017) *1997 vs. 2007 vs 2017: how is videogame character design evolving*. Available from: <https://www.pcgamesn.com/videoga-me-character-design-part-one> [Accessed 1st September 2022]

Kuntjara, H. & Almanfaluthi, B. (2017) *Character Design in Games Analysis of Character Design Theory*. Available from: <https://journal.binus.ac.id/ind-ex.php/jggag/article/view/7197/4083> [Accessed 26th July 2022]

Lankoski, P. (2002) *Character Design Fundamentals for Role-Playing Games*. Available from: https://www.researchgate.net/publication/200010276_Character-_Design_Fundamentals_for_Role-Playin-g_Games [Accessed: 13th August 2022]

Rajpurohit, P. (2022) *3D Character Modeling for Games: The Detailed Guide*. Available from: <https://www.mindinventory.com/blog/3d-character-modeling-for-games> [Accessed 29th July 2022]

Unreal Engine. (2022) *MetaHumans Documentation*. Available from: <https://docs.metahuman.unrealengine.com/en-US/mesh-to-metahuman-quick-start> [Accessed 27th August 2022]



© 2022. Authors. Published by the University of Novi Sad, Faculty of Technical Sciences, Department of Graphic Engineering and Design. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution license 3.0 Serbia (<http://creativecommons.org/licenses/by/3.0/rs/>).

