



Proposed workflow for UV mapping and texture painting

Mostafa Hassan

This thesis is submitted to the Faculty of Computing at Blekinge Institute of Technology in partial fulfillment of the requirements for the degree of Bachelor of Science in Digital Game Development. The thesis is equivalent to 10 weeks of full time studies.

Contact Information:

Author(s):

Mostafa Hassan

E-mail: mohc12@student.bth.se

University advisor:

Francisco Lopez

Department of creative technologies

Faculty of Computing
Blekinge Institute of Technology
SE-371 79 Karlskrona, Sweden

Internet : www.bth.se
Phone : +46 455 38 50 00
Fax : +46 455 38 50 57

ABSTRACT

Context. There are several workflows for texturing 3D models. 3D models will often have to be constructed and textured before they can be viewed in a game engine. Files would have to be exported and imported in order to view the result.

This thesis will look at the usability of having the programs that are used to construct and texture assets connected with each other. In other words, a program would send and receive data in real-time which can be used to avoid the exporting and importing of assets.

Objectives. Define a better workflow for texturing models that will be used for games. Compare the usability in terms of speed and the bother of managing asset files.

Methods. This work utilizes a comparative experiment where subjects get to test and evaluate two workflows, the traditional workflow which the subjects should already be familiar with, and the prototype system that allows subjects texture painting in real-time.

Results. Results showed that all participants conducted the experiment faster using the proposed workflow rather than using the traditional one. According to the questionnaire, participants preferred the proposed workflow and did not mind having multiple application windows open simultaneously.

Conclusions. The results of both questionnaire and the time correlation data were positive, suggesting that using real-time viewing when texturing assets can enhance efficiency.

Keywords: real-time texturing, workflow

CONTENTS

ABSTRACT	I
CONTENTS	II
LIST OF FIGURES.....	III
LIST OF TABLES.....	IV
1 INTRODUCTION	1
1.1 RELEVANCE	3
1.2 OBJECTIVE	3
1.3 RESEARCH QUESTION AND PROBLEM.....	3
2 RELATED WORK	5
2.1 TEXTURE MAPPING.....	5
2.2 ADAPTIVE UNWRAPPING	6
2.3 INTERPROCESS COMMUNICATION	7
3 METHOD	8
4 PLUGIN DESIGN.....	10
4.1 MAYA PLUGIN.....	10
4.2 GIMP PLUGIN.....	10
4.3 GAME ENGINE	11
5 EXPERIMENT	13
5.1 PREPARATION AND SETUP.....	13
5.2 PARTICIPANTS.....	13
5.3 QUESTIONNAIRE	13
5.4 TIME METRIC	14
5.5 EXPERIMENTAL DESIGN.....	14
6 RESULTS	15
6.1 QUESTIONNAIRE	15
6.2 TIME PERIOD.....	16
7 CONCLUSION AND FUTURE WORK	19
7.1 EVALUATION OF RESULTS	19
7.2 CONCLUSIONS.....	19
7.3 DISCUSSION	20
7.4 FUTURE WORK	21
APPENDIX A.....	24
APPENDIX B.....	25
APPENDIX C.....	27
APPENDIX D.....	33
REFERENCES	40

LIST OF FIGURES

Figure 1: 3D Cube model	1
Figure 2: Unwrapped model (cube) onto a 2D plane	1
Figure 3: The link between image manipulation program, 3D viewer and 3D program.....	2
Figure 4: Texture mapping a cube	5
Figure 5: Chameleon program.....	6
Figure 6: UV layout and diffuse texture from Chameleon.....	6
Figure 7: Exchanging data between two processes using inter-process communication.....	7
Figure 8: The model that will be used for the texturing task.....	9
Figure 9: UV layout constructed from a Glock pistol model	9
Figure 10: Maya plugin user interface.....	10
Figure 11: Highlight selected UV coordinates of polygon on the UV-layout	11
Figure 12: Pixels sent from GIMP to the 3D viewer	12
Figure 13: Workflow in action with GIMP's single-window mode	12
Figure 14: Formula for workflow (B) correlation	17
Figure 15: Formula used for calculating standard deviation	17
Figure 16: Average time for each workflow, with error bars	18

LIST OF TABLES

Table 1: Likert scale for questionnaire	13
Table 2: Multiple windows are not a distraction	15
Table 3: Workflow B preferred	16
Table 4: Participants time correlation for workflow A and workflow B	16
Table 5: Analysis of variance data.....	18

1 INTRODUCTION

There are several workflows for creating assets for games such as models, user interfaces (UI), head-up-displays (HUDs) etc. Assets need to be created before they can be viewed in a game engine. One method for texturing a 3D mesh, that is created in a 3D graphics software, is by projecting the mesh onto a 2D planar surface so the texture mapping process can be made [1], [2].

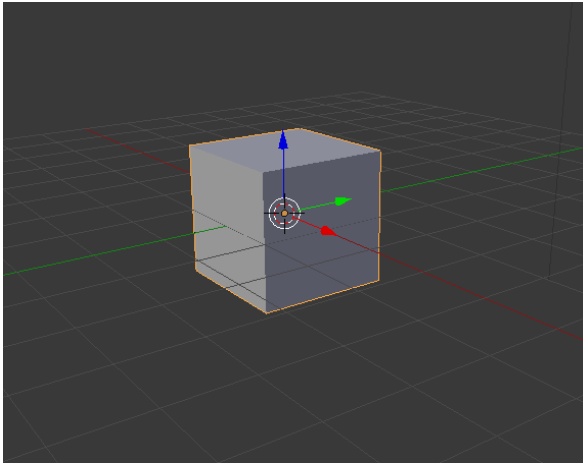


Figure 1: 3D Cube model

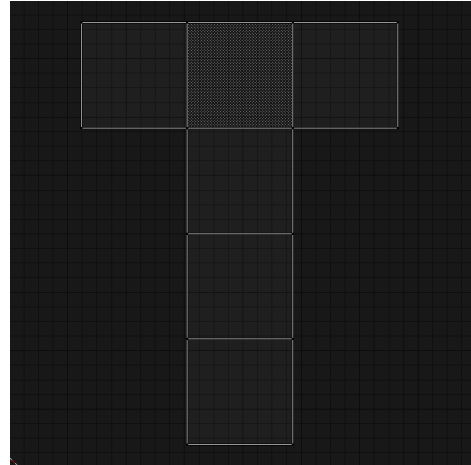


Figure 2: Unwrapped model (cube) onto a 2D plane

This process of projecting models to an UV (planar) space is called UV Unwrapping. An artist will often want to have a layout that makes it easy to distinguish parts from the mesh. Figure 2 is considered a good UV layout of a cube because of the effortlessness to piece together that it originates from Figure 1, a cube. The model is ready to be textured using an image manipulation program (IMP) such as Photoshop or GIMP, because it has been both modelled and unwrapped [3], [4].

Some modern game engines, such as Unity, support live updating of assets and allow for a quick reviewing process. In Unity, all assets that were changed from the outside (active window) update instantly after they have been both edited and saved.

Having an “offline” approach to texturing has some disadvantages:

- The exporting and importing of textures takes time, especially if the texture is big.
- An asset does not always look exactly the same in a 3D viewer as it does in the 3D graphics software, which often means a lot of retouching before the asset is ready.
- The texture is applied on a 2D projection of the mesh, which can result in some difficulty in knowing where the painting operations are applied on the model.
- An artist often has to export an UV layout to the image manipulation program in order to use it as a template. The template can be used to roughly

determine each part corresponding polygons in the layout for the object that is in question.

- The complexity of a model determines the difficulty of knowing which triangle is being affected by the paint.
- The artist would often have to save the texture and then reload it in the 3D graphics program in order to render the model and check whether it looks like it is supposed to.

In this work I will propose a workflow for UV mapping and texture painting models that will allow for rapid interaction between tools and live viewing of the result in a 3D viewer.

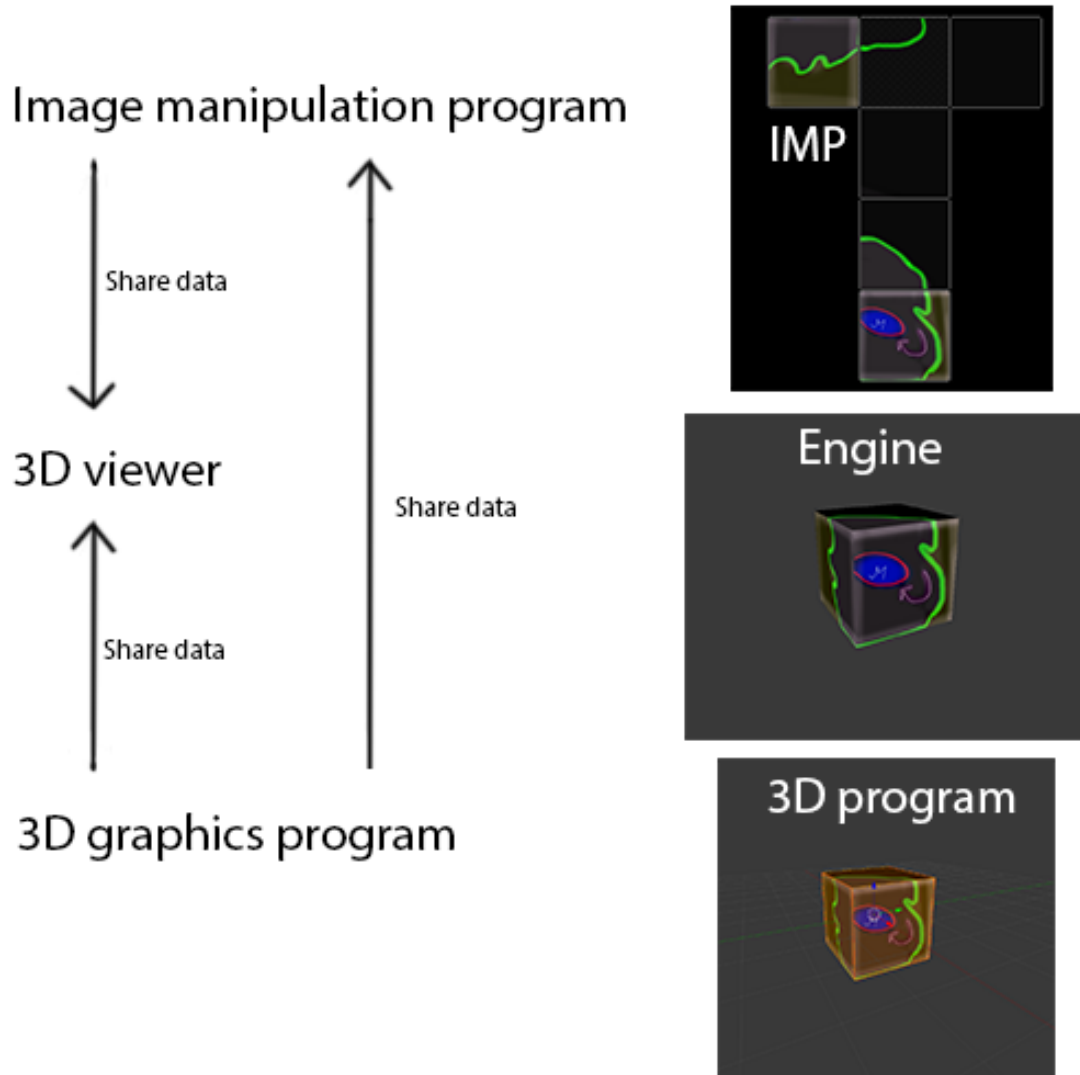


Figure 3: The link between image manipulation program, 3D viewer and 3D program

The workflow requires an IMP (image manipulation program), engine/viewer and a 3D program (Figure 3). The idea is to eliminate the need to manually save and load texture files by sharing “live” data between GIMP, Maya and a rendering or game engine. The communication will be done using Windows Filemaps [5] (shared memory between processes). The imaging program will share back and forth necessary data such as pixels in real time which the 3D viewer can use to view, rebuild and save. This will result in a real time viewing and exporting of a file that represents the actual model.

1.1 Relevance

When this idea was proposed to the supervisor and research was done on originality and contribution, a patent claiming a similar workflow found by, S. Prahalad and S. Chogle (NVIDIA) [6]. This patent has not been granted yet, but it gives us an indication of the relevance of the problem and the significance of the contribution by implementing such a system and also by performing the user studies.

Modern 3D programs such as Blender 3D allow for users to paint directly on the mesh. This means that no external tools will be needed as constantly. These 3D programs can introduce a certain degree of a learning curve but have no advanced features like industry established applications such as GIMP and Photoshop when it comes to texture painting [7].

Furthermore, the development of such a tool gives the opportunity to display the mix of knowledge of a TA (technical artist). An ordinary computer programmer can show skills by delivering a code snippet or such and a regular graphical artist can show some art in a portfolio of some sorts. A TA is the link between the programmer and the graphic artist which means showing qualities from a TA can be a bit harder.

Nonetheless, a good way to show qualities of a TA is to create a tool that combines knowledge from both programmers and graphic artists. A real-time texturing tool does that since it needs someone that knows how texturing works in order to create a plugin that will enable graphic artists to texture more efficiently.

1.2 Objective

As presented before, a large amount of exports and imports must be made for each asset using the traditional workflow. This thesis proposes to connect the programs that users use for modelling and texturing with the intent of making it possible to work more efficiently. The focus of this project will be to enhance the traditional workflow by defining a workflow that is based on the traditional workflow by using existing tools and programming techniques acquired during the Technical Artist (TA) program at Blekinge Institute of Technology. Comparing performance and the bother of managing multiple application windows is also an objective.

1.3 Research Question and problem

The questions this thesis will address are:

1. Would the user of the proposed workflow be able to perform a task in a shorter time period rather than using a traditional one?
2. The proposed workflow requires multiple software windows running at real-time; will the “visual aid” help the user whilst texturing, or will it introduce a distraction?

- 1.3.1 Would the user of the proposed workflow be able to perform a task in a shorter time period rather than using a traditional one?

The purpose of the comparison is to see if the suggested workflow would be preferred over the traditional workflow because of the usability improvement; measuring efficiency can be done by measuring time.

- 1.3.2 The proposed workflow requires multiple software windows running at real-time; will the “visual aid” help the user whilst texturing, or will it introduce a distraction?

A distraction can be introduced since the proposed workflow requires multiple application windows running at the same time. GIMP, Maya and a 3D viewer on two screens may not be ideal, however if the user is finished with the modelling and UV unwrapping part, Maya will not be as necessary to see anymore which means that its window could be minimized if so desired.

1.3.3 Definition of Usability

This project will evaluate the usability of two workflows. Generally, the aspects to focus on are:

- **Learnability**
The simplicity to accomplish basic tasks the first time a user runs into the design.
- **Efficiency**
How fast users can perform a task once they have learned the design.
- **The time a task requires and errors**
How fast it takes to learn the design and how many errors users make.
- **Users’ subjective satisfaction.**
How pleasant it is to use the design.

The usability metrics above can be a big impact on how fast a user can perform a task. It may be easy to specify usability metrics, but it is slightly difficult to collect them. However, this study will mostly focus on the efficiency part since that is what is going to be the answer to the research questions. In other words, each participant will perform tasks whereas the research leader keeps track of the time that has passed [8], [9], [10].

2.1 Texture Mapping

Texture mapping is a technique for defining detail or colour on a model. Initially, it was a technique that was only supposed to wrap mapped pixels from a texture to a 3D surface. It is called diffuse mapping nowadays to differentiate it from other mappings such as bump mapping, normal mapping and other complex variations on the technique. The key point to remember is that it does not rely on having to increase the geometric resolution. The data can be stored in various ways, for instance, a simple one dimensional array or a complex three dimensional array depending on what the texture is going to be used for. The technique is really powerful and beneficial, particularly with models that are going to be used for games considering that no extra polygons would have to be added [11], [12].

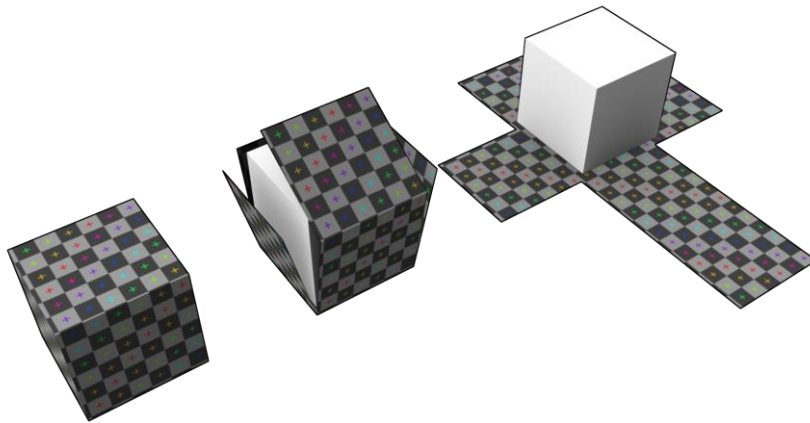


Figure 4: Texture mapping a cube

A diffuse map typically contains RGB (red, green, blue) colour data. Occasionally it contains an additional channel for alpha blending (RGBA). When a model is transformed, the UV coordinates are typically reformed with the vertices as well, which delivers the necessary data for precise illustration of a texture regardless of the transformation of a model. Generally, a model is characterised in object space as a sequence of vertices with complementary UV coordinates. Using various procedures for interpolating UV coordinates before rasterization of the final image and map pixels to these UV coordinates will provide a complete UV space of the surface in question [11], [12]. See Figure 4.

2.2 Adaptive Unwrapping

The closest approach to the proposed workflow found in the literature is an Adaptive Unwrapping paper by T. Igarashi and D. Cosgrove [13], and although it has a similar approach to the problem, it lacks the benefits of using full-featured and industry established applications like Maya 3D and GIMP to do the tasks of modelling and texture painting.

There are pros and cons with the adaptive unwrapping technique. The pros being that developers have full control over the whole texturing pipeline, due to having their own painting software, which could benefit users in terms of workflow and user experience, as well as having more efficient and optimized data transfer during the pipeline process. The cons being that the program lacks advanced features which will limit advanced users. The assets will have to be exported from the Chameleon program (the program that Igarashi and D. Cosgrove team created) and imported into a game engine in order to be viewed.



Figure 5: Chameleon program.

Image downloaded from
[http://www-ui.is.s.u-
tokyo.ac.jp/~takeo/research/chameleon/chameleon.htm](http://www-ui.is.s.u-tokyo.ac.jp/~takeo/research/chameleon/chameleon.htm)
in 31-May-2016

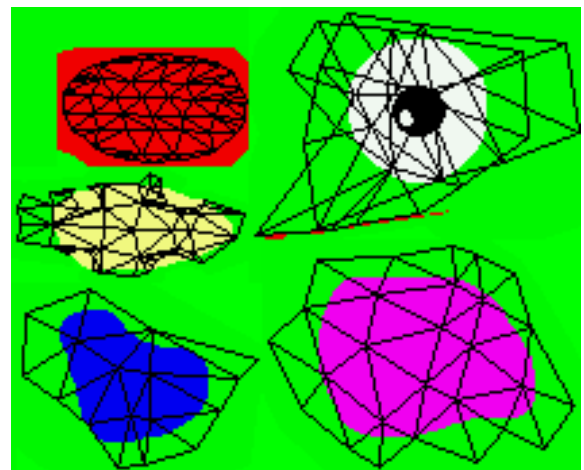


Figure 6: UV layout and diffuse texture from

Chameleon. Image downloaded from [http://www-
ui.is.s.u-
tokyo.ac.jp/~takeo/research/chameleon/chameleon.htm](http://www-ui.is.s.u-tokyo.ac.jp/~takeo/research/chameleon/chameleon.htm)
In 31-May-2016

Figure 5 displays a simple mesh that has been textured in the Chameleon program and Figure 6 displays the UV layout that the program constructs using mesh data. It is clear that the program is innovative and useful; however, the program might only be useful when conducting basic texturing tasks.

The main difference between the proposed workflow and the adaptive unwrapping workflow is that the proposed workflow uses industry established applications such as Maya and GIMP, which allows users to use advanced features in programs that they have probably tried or heard of. Another key difference being that the proposed workflow depends on having more than one window open simultaneously whilst working. Finally, the proposed workflow allows textures to be applied directly on the 3D model and viewed in the 3D viewer in real time, which will not require importing or exporting of any kind.

2.3 Interprocess Communication

Interprocess Communication (IPC) is used to exchange data between two or more separate processes (running programs). Some operating systems, such as Windows, provide facilities for IPC. Naturally, applications can use IPC categorized as clients and servers. Some applications are both clients and servers which means that they can both write and read data [14].

There are various IPC methods that can be used for applications, one of them being file mapping, which essentially allows a process to treat the information in a file as if they were a memory block in the processes address space. Each process receives a pointer to the memory, which can be used to read or change the contents of the file in question. This method is quite effective and it provides some operating-system-supported security. It is important to recognize that this method can only be used between processes on a local computer; with this in mind, file mapping cannot be used over a network [14].

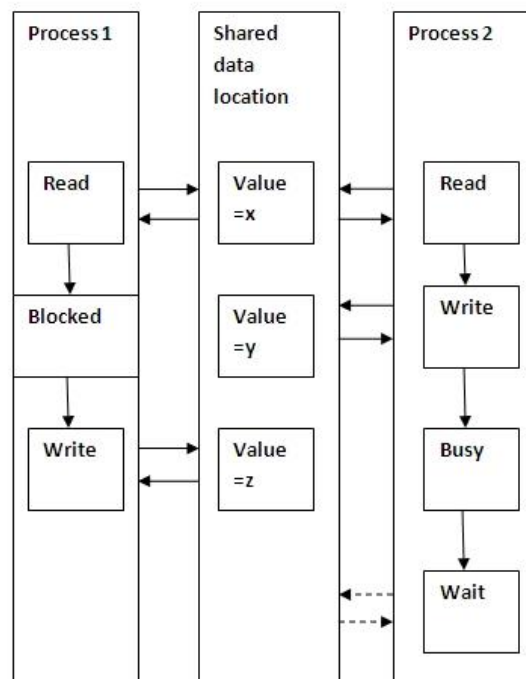


Figure 7: Exchanging data between two processes using inter-process communication.
Image downloaded from <http://www.moreprocess.com/process-2/race-condition-in-inter-process-communication-ipc> in 31-May-2016

Synchronization is an important part of IPC whether it is done by using algorithms or mutex locks to make clients and servers collaborate. A mutex is used to “lock” the memory for whole transactions, in other words, until a component of the processes (thread) that owns the mutex has accessed the memory and executed some code. A signal is sent to a mutex object when it is not owned by any thread, clients and servers collaborate by preventing the other from reading and writing to shared memory at the same time, which means that each thread would have to wait for ownership of a mutex before executing the code that accesses the memory (Figure 7) [15], [16].

3 METHOD

The key components of the work will be the implementation of the plugins for GIMP and Maya that will allow the programs to communicate with each other. The other key component is an experimentation to test the efficiency compared to traditional methods.

An UV layout will be constructed and displayed, using the vertices from the model that GIMP receives from Maya. The plugin will also highlight UV coordinates in GIMP that are selected in Maya.

A basic 3D viewer will also be implemented; the viewer will be thought of as a game engine in order to see and comprehend the possibilities with using the proposed workflow. Visual studio with C++ and DirectX will be used to develop the viewer application. In order to be able to optimize the plugin if needed, Maya API with C++ will be used to create the Maya plugin.

The plugins will be evaluated and tested by participants in a comparative experiment using a questionnaire. The experiment will measure how fast a user can perform a texturing task for both workflows.

Participants will receive an introduction to what the experiment will measure. A pilot test will be provided for each participant at the start, the test consists of a basic texturing task where the participant will have to texture a regular cube by following instructions. Screenshots will be included in the instructions in order to make it as clear and logical as possible for the participants.

The purpose of the pilot test is to help the participants get acquainted with the proposed workflow and with how typical instructions are formulated. Once the pilot test is finished, the real texturing task begins where measuring time will be imperative. The texturing task will have to be done with both workflows before the questionnaire can be answered. Additionally, each participant would also have to sign a consent form that will be provided before the experiment starts.

Not all participants will start with conducting the texturing task using the traditional workflow, which will be called workflow A in this thesis. It was decided that every other participant will start with conducting the texturing task using the proposed workflow (workflow B) instead in order to get as accurate data as possible. The reason that the data would be more accurate is because the time for workflow B would probably always be less than workflow A if the participant started with workflow A since the participant would have already been familiar with the texturing task and with what techniques that works best for completing the task in question.

The diffuse texture will have to be applied to a complex model in order to really comprehend the possibilities for such a workflow. However, it is just as important to texture something that would be considered a real texturing task that a graphical artist would be doing for a game industry. For this reason, the texturing task will consist of subtasks where each subtask would be to texture a small part of a 3D

weapon. The weapon that will be used is a Glock pistol (Figure 8) taken from the game called Counter-Strike: Global offensive (GO). There is a market for global offensive skins (textured weapons for the game). Online shops, forums and competitions exist for buying, discussing and creating global offensive skins [17], [18], [19]. For this reason, a pistol model like this would be a great challenge especially since it can be somewhat complex in its shapes and UV coordinates (Figure 9).



Figure 8: The model that will be used for the texturing task. Image downloaded from <http://counterstrike.wikia.com/wiki/Glock-18> in 31-May-2016

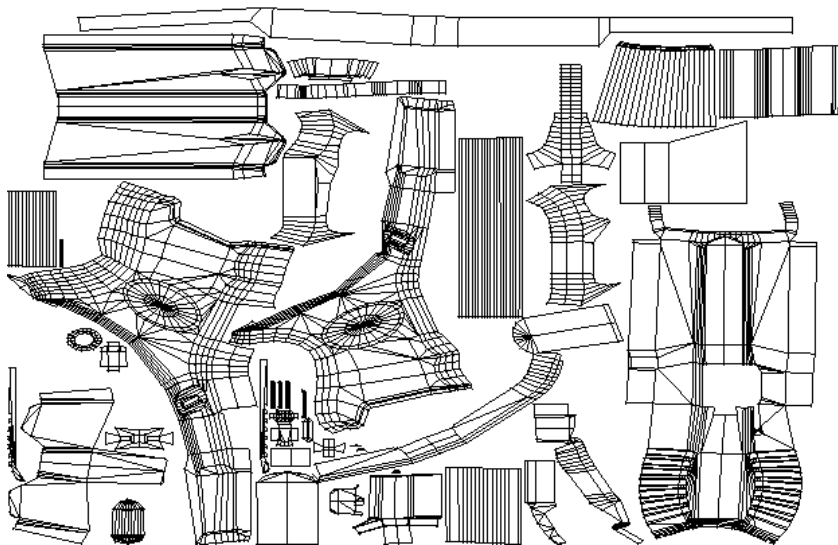


Figure 9: UV layout constructed from a Glock pistol model

A participant would have to be familiar with the traditional way of texturing in order to be able to participate in the experiment. That said, each participant would have to have some experience in 3D modeling programs and image manipulation programs. All participants received a document with questions such as age and whether they were familiar with traditional way of texturing that they had to answer in order to be certain that the participant had previous experience with texturing and that they would be able to finish the experiment.

4.1 Maya plugin

Maya will act as a controller because the 3D viewer and GIMP need data from Maya. Maya will also be used to construct and unwrap 3D models and to select different parts of the models in order to have their UV coordinates sent to GIMP to be fabricated.

The plugin will be designed to effortlessly start GIMP, and a 3D viewer. With just one click, the user will immediately be able to start painting and viewing the final results in real time. The plugin will offer some important options such as being able to send selected UV coordinates of polygons to GIMP (Figure 10). Options that are less important but still supportive such as being able to not send the UV layout will also be available. The user could for instance choose to only send the selected polygons instead of sending both UV layout and selected UVs.

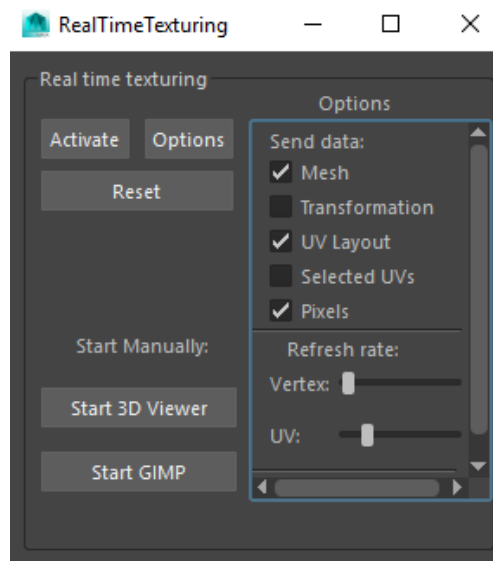


Figure 10: Maya plugin user interface

4.2 Gimp plugin

Gimp will be used to draw the diffuse texture for the model and to share the texture with the 3D viewer in order to apply it to the 3D model. Data will be received from Maya, such as UV coordinates and options. It is essential that GIMP takes these coordinates and options into account when constructing the UV layout. There is a noticeable delay when GIMP is generating the UV layout, especially if the model has more than one thousand polygons.

The plugin will create a new layer and draw each line from the UV coordinates list automatically on that layer. Furthermore, data such as UV coordinates will have to be retransmitted every time the user modifies that data.

A new layer for highlighted lines will be created as well. Each triangle will be highlighted in GIMP in order for the user to be able to find out where each polygon is on the layout.

The plugin transfers pixels to the 3D viewer as fast as it can, however it still takes time depending on the size of the texture and computer speed. Texture data is transmitted and viewed in the 3D viewer every 1.5 seconds for a 1024x1024 texture using an Intel i5 processor. Importantly, pixel transference speed does not depend on the complexity of the model, with this in mind, texture data will be updated and viewed in the 3D viewer every 1.5 seconds whether the model is complex or not.

4.3 Game Engine

The 3D viewer has to be thought of as a game engine in order to recognize the potential of this workflow; the user would be able to see the assets with the environment of the game in real time.

High dynamic range imaging (HDRI), shadows and reflections are just examples of what can affect the assets. The viewer that was created for this work was just a 3D viewer with no options for HDRI or shadows. It is only used to show the basics of the workflow.

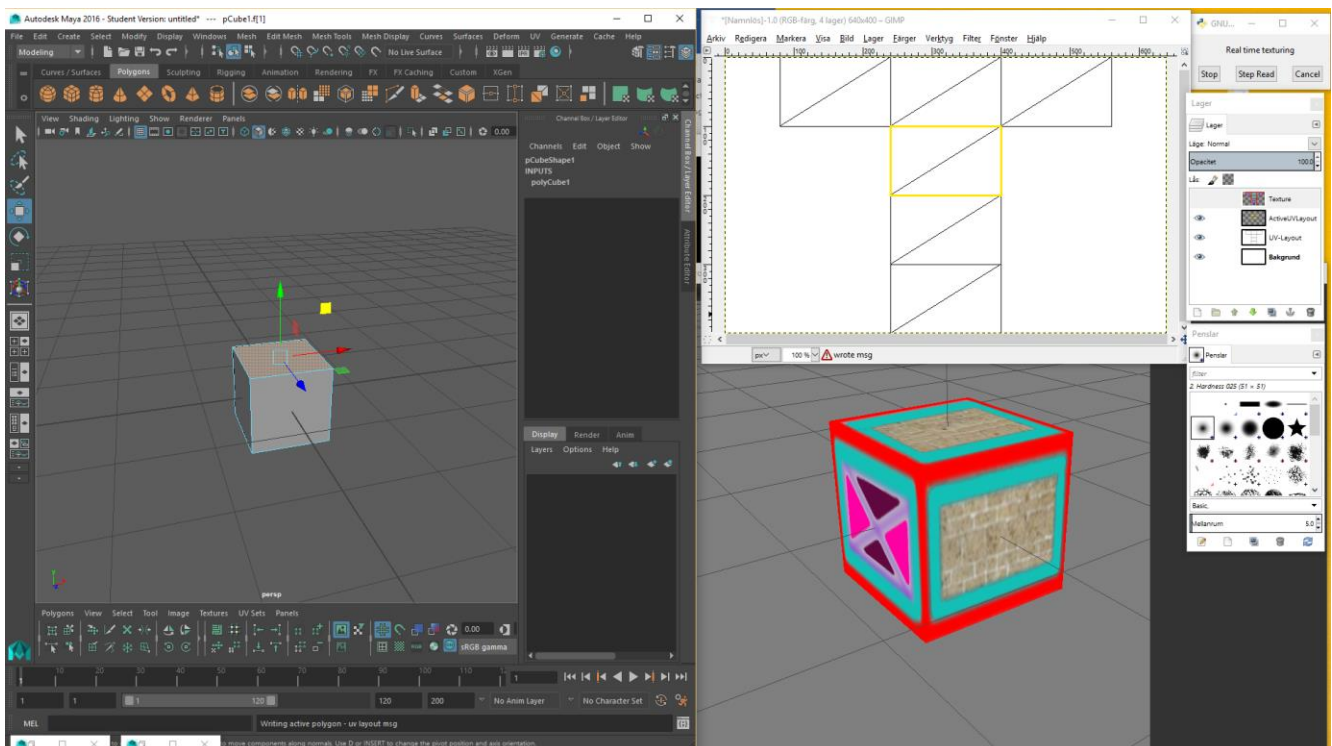


Figure 11: Highlight selected UV coordinates of polygon on the UV-layout

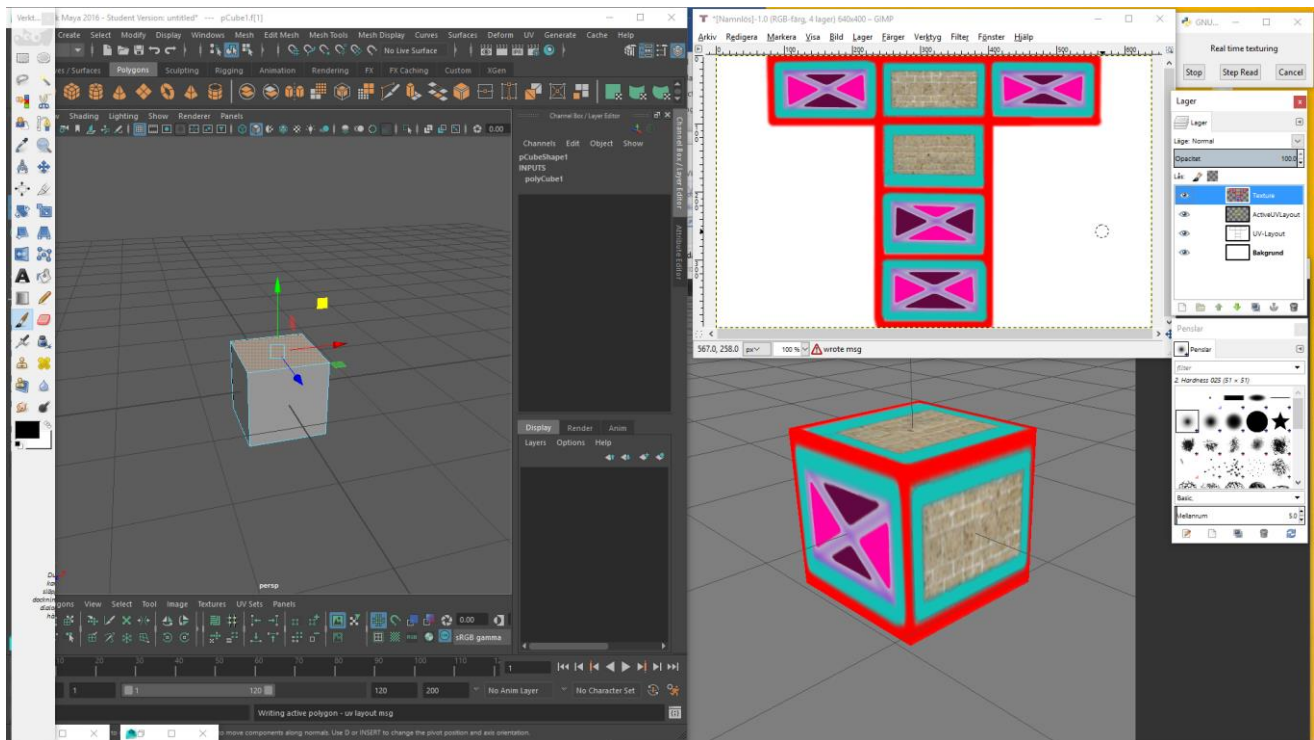


Figure 12: Pixels sent from GIMP to the 3D viewer

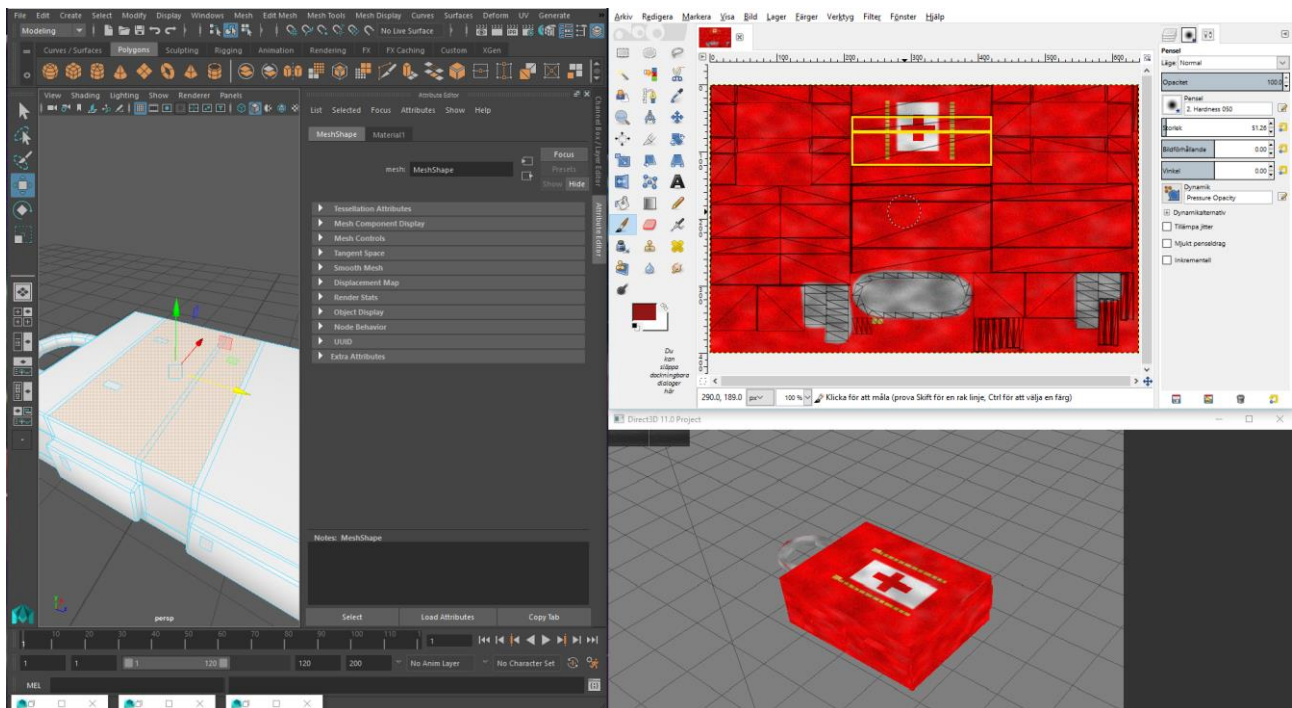


Figure 13: Workflow in action with GIMP's single-window mode

Figure 11 and Figure 12 displays how working with the proposed workflow using only one screen and GIMP's multi-window mode can be a bit too much for the eyes if the user does not have a big screen, even when only working with a texture size of 640x400 pixels. Using GIMP's single-window mode (Figure 13) can perhaps make it less confusing.

5.1 Preparation and Setup

The experiment was conducted in a private room where there were no distractions. Access to the room was controlled to evade unpredicted disturbances throughout the experiment. The room was prepared with a HP ENVY h8-1514eo computer that has an Intel i5 processor, 12 GB RAM and a NVIDIA GeForce GTX 660 graphics card. Two 24-inch screens were used for the experiment. GIMP (single-window mode), Maya, and a viewer were prepared on the computer before the experiment started.

5.2 Participants

The subjects were mostly students from Törnströmska high school; however, some had finished their high school studies. There were four students from Blekinge Institute of Technology. A 3D graphics artist and educator participated in the experiment as well. Blekinge Institute of Technology has a technical artist program and the high school has a Game graphics, 3D and animation program. The students were in their second and third year of these programs, which means that they have acquired experience in modeling and texturing. Ages were between 16 and 45 with the average age being 22. There was a total of 18 subjects participating in the experiment, 16 males and 2 females. All subjects that were chosen had previous experience in texturing with the traditional workflow and 3D modeling programs.

5.3 Questionnaire

A decision was made to use closed format and Likert questions in order to straightforwardly and effortlessly find out which workflow was preferred and whether the visual aid was a distraction or not. All participants answered the same questionnaire. See Table 1.

The Likert [20] questions will help establish how strongly the participants agree to a particular statement, in this case whether having multiple windows opened simultaneously was a distraction or not, by rating their level of agreement or satisfaction from 1 to 5 [21].

- I experienced a distraction from having multiple windows opened simultaneously.

Table 1: Likert scale for questionnaire

Strongly Disagree	Disagree	Undecided	Agree	Strongly Agree
(1) <input type="checkbox"/>	(2) <input type="checkbox"/>	(3) <input type="checkbox"/>	(4) <input type="checkbox"/>	(5) <input type="checkbox"/>

The closed format will allow multiple choice questions, where participants are restricted to choose among any of the given choice answers [22].

Question from questionnaire:

- Which workflow would you prefer, workflow A (traditional workflow) or Workflow B (real time texturing and viewing workflow)?

Workflow A ☐ Workflow B ☐

5.4 Time Metric

All subjects did not start with workflow A. It was decided that every other subject would start with conducting the experiment using workflow A, the other subjects would instead start with workflow B. The lead researcher would start taking time when the subject started with the texturing task [23] and stop taking time when the participant had painted the last stroke for the last subtask. The time will be reported in three ways:

1. **Average task completion time**

This is the most common way to report time. It includes only the users who finished the task successfully.

2. **Mean time to failure**

This section is the average time subjects spent on the task before they gave up or completed the task inaccurately.

3. **Average time on task**

This section is the total duration the users spent on each task.

5.5 Experimental Design

The experiment consisted of the following steps:

Consent form The participant was given a consent form, informing that the contribution is voluntary, confidential and anonymous. It also states that there is permission to administer the results of the survey.

Introduction Inform contributors, in writing or verbally if desired, how the experiment can be accomplished and what metrics are going to be measured. See [Appendix B](#).

Pilot Experiment This pilot experiment will prepare the participant for the actual experiment and validate if it is feasible for the participant to complete the experiment. Every other participants begins to conduct the pilot test for workflow A ([Appendix C](#)), the second begins with workflow B instead.

Test workflows The subject tests the workflows by accomplishing a texturing task just like in the pilot experiment, the difference in this section is that the lead researcher will measure time whilst the participant is doing the task. Participants will start with the same workflow order as they did in the pilot experiment [24]. See [Appendix D](#).

Questionnaire In this section, the participant answers the provided questionnaire.

6.1 Questionnaire

The questionnaire started with a statement (Table 1) instead of a question; participants could rate their level of agreement or disagreement about whether it was a distraction to have multiple application-windows open simultaneously or not. Table 2 shows an overview of the results from the statement. The statement read as follows: I experienced a distraction from having multiple windows opened simultaneously.

➤ **Table 2:** I experienced a distraction from having multiple windows opened simultaneously

Level of agreement or disagreement	N	%
Strongly Disagree	6	30%
Disagree	9	50%
Undecided	3	20%
Agree	0	0%
Strongly Agree	0	0%

The Likert scale table shows that 6 participants (30% of all participants) did not think that having multiple application windows opened simultaneously was a distraction. 9 participants (50% of all participants) thought that it was not a problem having the windows there and 3 participants (20% of all participants) were undecided.

The study revealed mild disagreement regarding the statement of whether it was a distraction to have windows opened simultaneously or not; mean value was calculated to be 1.8. The formula that was used to calculate the mean value: [(number of people who selected response 1) * (increment of response 1) + (number of people who selected response 2) * (increment of response 2)... (number of people who selected response n) * (increment of response n)] / (total number of respondents) [21], [25], [26].

Mean value calculation:

$$((6*1) + (9*2) + (3*3) + (0*4) + (0*5)) / 18 = 1.83$$

A question where it was possible to discover which workflow each participant preferred was a part of the questionnaire as well. Each participant could only answer “workflow A” or “workflow B”. The question read as follows: Which workflow would you prefer, workflow A (traditional workflow) or Workflow B (real-time texturing and viewing workflow)?

Table 3 shows an overview of the results from the question.

Table 3: Workflow B preferred

Workflow	N	%
Workflow A (traditional workflow)	0	0
Workflow B (proposed workflow)	18	100%

As Table 3 demonstrates, there is a preference towards the proposed workflow implementation. The three columns consist of data collected from the questionnaire, where the first column consists of different workflows and the second of amount of participants that nominated the workflow in question. The last column shows the portion of participants that voted for a specific workflow. Data revealed that 100% of the participants preferred workflow B over workflow A.

6.2 Time period

Time was taken for each task that the participant conducted using both the traditional workflow and proposed workflow in order to find out if users could perform a task in a shorter time period using the proposed workflow rather than a traditional workflow.

Average task completion time includes the participants who finished the experiment successfully. The following diagrams and tables demonstrates an overview of difference in terms of speed between workflow A and workflow B.

Table 4: Participants time correlation for workflow A and workflow B

Participant	Workflow A (traditional workflow) time	Workflow B (Proposed workflow) time	Workflow (B) correlation	Workflow B faster than A
Participant 1	10.17	05.47	54%	46%
Participant 2	11.02	08.42	76%	24%
Participant 3	10.22	07.35	72%	28%
Participant 4	10.13	08.02	79%	21%
Participant 5	13.33	09.21	69%	31%
Participant 6	12.31	07.35	60%	40%
Participant 7	11.09	06.21	56%	44%
Participant 8	14.03	08.20	58%	42%
Participant 9	13.08	10.03	77%	23%
Participant 10	14.05	10.36	74%	26%
Participant 11	13.42	11.19	83%	17%
Participant 12	15.04	11.06	74%	26%
Participant 13	13.31	09.51	71%	29%
Participant 14	14.13	11.37	80%	20%
Participant 15	11.51	06.26	54%	46%
Participant 16	14.05	11.17	80%	20%
Participant 17	15.51	13.27	86%	14%
Participant 18	14.53	12.47	86%	14%

Table 4 presents the time correlation between the workflows; the second and third columns displays how fast (in minutes) each participant (first column) conducted the task when using workflow A and when using workflow B. It is very clear that all 18 participants were able to perform a task in a shorter time period using workflow B rather than using workflow A.

The forth column shows that the participant in question completes the task using workflow B in x% of the time it takes for the participant to do it using workflow A. This was calculated by dividing the time for workflow B with the time for workflow A and multiplying that value with a hundred to get the result in percentage. See Figure 14.

$$\frac{\text{Workflow B time}}{\text{workflow A time}} \times 100\%$$

Figure 14: Formula for workflow (B) correlation

The last column displays how much faster workflow B was compared to workflow A for each participant. A chart can be viewed, that displays each participants time for both workflows. See [Appendix A](#).

The average value from the extracted data (last column) is a 28% increase in terms of speed for workflow B compared to workflow A. The values where it was displayed workflow (B) correlation (forth column) have a standard deviation of 0.11 and an average of 72%.

The standard deviations are 1.7 for workflow A and 2.3 for workflow B. Average time for workflow A is 12.8 minutes whilst 9.3 minutes for workflow B.

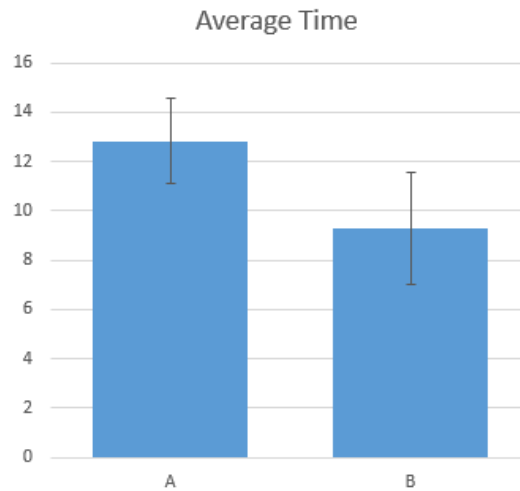
The standard deviation was calculated by first calculating the mean and then for each value subtracting the mean and squaring the result. The mean was then worked out from those squared differences. Lastly the result was given by taking the square root. See Figure 15.

$$\sigma = \sqrt{\frac{1}{N} \sum_{i=1}^N (x_i - \mu)^2}$$

Figure 15: Formula used for calculating standard deviation

Figure 16 shows the average time for each workflow, with standard deviation error bars, where A stands for workflow A and B for workflow B; the numbers on the vertical axis is still in minutes.

Figure 16: Average time for each workflow, with error bars



Single Factor ANOVA was applied to analyze the performance variation among the two workflows. A significant variation was found between workflow A and workflow B with regard to the task completion times for the distribution of scores (editing mask). See Table 5.

Table 5: Analysis of variance data

SUMMARY						
<i>Groups</i>	<i>Count</i>	<i>Sum</i>	<i>Average</i>	<i>Variance</i>	<i>Std</i>	
Workflow A	18	230.93	12.82944	3.026382	1.73965	
Workflow B	18	166.92	9.273333	5.096682	2.257583	
ANOVA						
<i>Source of Variation</i>	<i>SS</i>	<i>df</i>	<i>MS</i>	<i>F</i>	<i>P-value</i>	<i>F crit</i>
Between Groups	113.8133	1	113.8133	28.02227	0.00001	4.130018
Within Groups	138.0921	34	4.061532			
Total	251.9054	35				
Hypothesis: $H_0: \mu_A = \mu_B$ $H_1: \mu_A \neq \mu_B$						

As can be seen in Table 5, F (28.02227) is more than Fcrit (4.130018) and the P value (0.00001) is really small, meaning that the average time it takes for each workflow are indeed significant different. In terms of performance, the proposed workflow was significantly better than the traditional workflow.

All 18 participants were able to successfully complete the experiment accurately. That said, there was no mean time to failure data to be analyzed or displayed.

Since there was no mean time to failure data to be analyzed the average time on task is the same as the average task completion time; this data is displayed in Table 4.

7.1 Evaluation of results

In this study, we have formulated two research questions with the intent of finding out if the proposed workflow has any potential. One question for performance and one for visual distractions; we have collected the user data through experiments, and performed analysis to answer the research questions.

The average time chart shows that there is a difference between the two workflows; results show that the real-time texturing and viewing method used in workflow B is more efficient in terms of speed even though having multiple application windows open simultaneously.

According to the results from the questionnaires, workflow B is more suitable when texturing models and therefore much preferred over workflow A. One participant said: “The method in workflow B was better suited since it eliminates the need of exporting and importing but also because it was easy to find out where on the model the texturing was inaccurate.”

Having multiple windows up simultaneously is no distraction according to the questionnaire since 50% of the participants totally disagreed with the statement and 30% casually disagreed. In conclusion 80% of the participants at least disagreed with the statement, which means that having multiple application windows open simultaneously is not a distraction. However, the participants used two screens when they conducted the experiment. With this in mind, having multiple application windows open simultaneously might be a distraction whilst texturing if the user only uses one screen. Workflow B may not be preferred over workflow A if the user only has one screen.

All things considered, it seems reasonable to assume that there is room for improvements for the proposed workflow when it comes to learnability, efficiency and subjective satisfaction.

7.2 Conclusions

Results of questionnaire, time correlation and analyses of variance data were very positive, suggesting that using real-time viewing when texturing assets can enhance efficiency fairly much.

To answer whether a user can perform a task in a shorter time period using the proposed workflow rather than using a traditional workflow, all 18 participants conducted the tasks faster using the proposed workflow (workflow B) and according to the analysis of variance data, the average time it takes for each workflow are significantly different. In conclusion, all data collected and analyzed suggests that the

workflow has a lot of potential and that a user can indeed perform a task in a shorter time period using the proposed workflow.

According to the questionnaire statistics, the majority of participants (80%) did not see the “visual aid” as a distraction whilst texturing which answers the second research question, whether the “visual aid” introduced a distraction or not.

7.3 Discussion

The development of the plugins seems to reveal the skill set of a technical artist since the plugins are created using artistic understanding of texturing and technical understanding of plugin construction.

The delay of creating layouts using UV coordinates can have affected the time correlation results and it could also have been a source of irritation for participants. If this experiment would have been repeated with the same workflow but no delays, the results may have differed from the results that have been presented.

Participants could also have been selected in a different way. As mentioned in section [5.2](#), all participants claimed that they were familiar with the traditional way of texturing and with 3D programs. But 95% of the participants had experience only from being a student at BTH or high school that offered 3D modeling and texturing programs. If other participants, for instance, people who have more experience within texturing and have been in the gaming business for several years were to perform the experiment, results may have differed from the results that have been presented.

Most participants had only the experience that they received from the 3D graphics and animation program in high school, which means that those participants only had about three years of experience. It is important to recognize that participants who only went to high school do not have the same experience that participants who went to BTH have. High school students do not have the same requirements as BTH students and they do not learn and work in the same tempo as university students do either.

As mentioned in section [5.5](#) not all participants started with workflow A, it was decided that every other participant would start with workflow B instead in order to get as accurate data as possible. The results would probably have been a bit different if all participant started with workflow B. Using another experimental design method would probably have differed the results as well.

Each participant textured the same model twice which means that the participant was already familiar with the topology of the model and with the UV layout. Using a different model that is exactly same in terms of complexity of both topology and UV layout could have given us a completely different result as well.

Average time could have differed if texturing tasks were more sophisticated and time consuming. However, it was important to have texturing tasks that were feasible to complete in the time that was given.

After an experiment was completed and the participants had filled out the questionnaire, the opportunity for an open discussion between the lead researcher and the participant opened up. Some of the participants asked if the proposed workflow would work with any 3D model and image resolution. The answer was that indeed the workflow could be used for any model but that the more complex the model is the longer it will take for the image manipulation program to create the UV layout. With this in mind, it might not be clever to use this technique, as it is, when texturing a high resolution and complex model as it will reduce performance and therefore efficiency as well. Furthermore, delay of pixel-transference will differ depending on the resolution.

The delay increases exponentially depending on the image size, a size of 1024x1024 pixels will take about 1.5 seconds to transfer and a 2048x2048 will take about 3.5 second to transfer. However, improvement could be made to the plugins in order to reduce delay time for both the construction of UV layout and the transference of pixels. Creating the UV layout as an image and compressing the image before sending it to GIMP and transferring only dirty tiles of an image instead of the whole image are examples on how to improve the plugins.

It was not expected that having multiple application windows open simultaneously would result in improving the workflow without causing distractions, but the results clearly show that it is no distraction at least whilst working on two screens. The results would probably have differed if the participants only used one 24-inch screen. However, it is common these days that an artist has two screens on a workstation. It is not certain if this workflow would be an increment when it comes to usability in terms of speed and whether all the windows would be a distraction or not when using this workflow on a one-screen workstation. It is in any case not expected that using one screen for workflow B would increase the usability.

7.4 Future Work

The experiment could benefit from having more participants with same backgrounds conducting the tasks. Having an experimental design were the pilot test is as complex as the real texturing task would perhaps be beneficial for a study like this. To that end, participants would be able to play around with the proposed workflow until they feel as comfortable with the techniques used when texturing a certain type of model as they do when using the traditional workflow, which would possibly result in more accurate data.

A couple of features that would also be interesting if this research was taken further would be to give the users the freedom to use tablets, phones with stylus or another computer such as a surface that could be used for the texture painting whilst having the 3D program and 3D viewer on the main computer. The difference would be that the programs would have to be connected via a fast LAN network (locally). This way of texturing would reduce the number of windows on the screen and probably increase comfort and user experience as well.

Interacting with the polygons of a 3D model in the game engine or directly in the image manipulation program. This would allow the user to select different parts of the model without having the 3D modelling software on screen all the time.

Viewing the changes without any delay would also increase comfort and user experience. There are two kinds of delays, one is the construction of the UV layout, and the other is the transference of pixels. Reducing delay time for the UV layout could be accomplished by creating the UV layout as an image in Maya and transfer it to GIMP instead of creating one triangle at the time in GIMP.

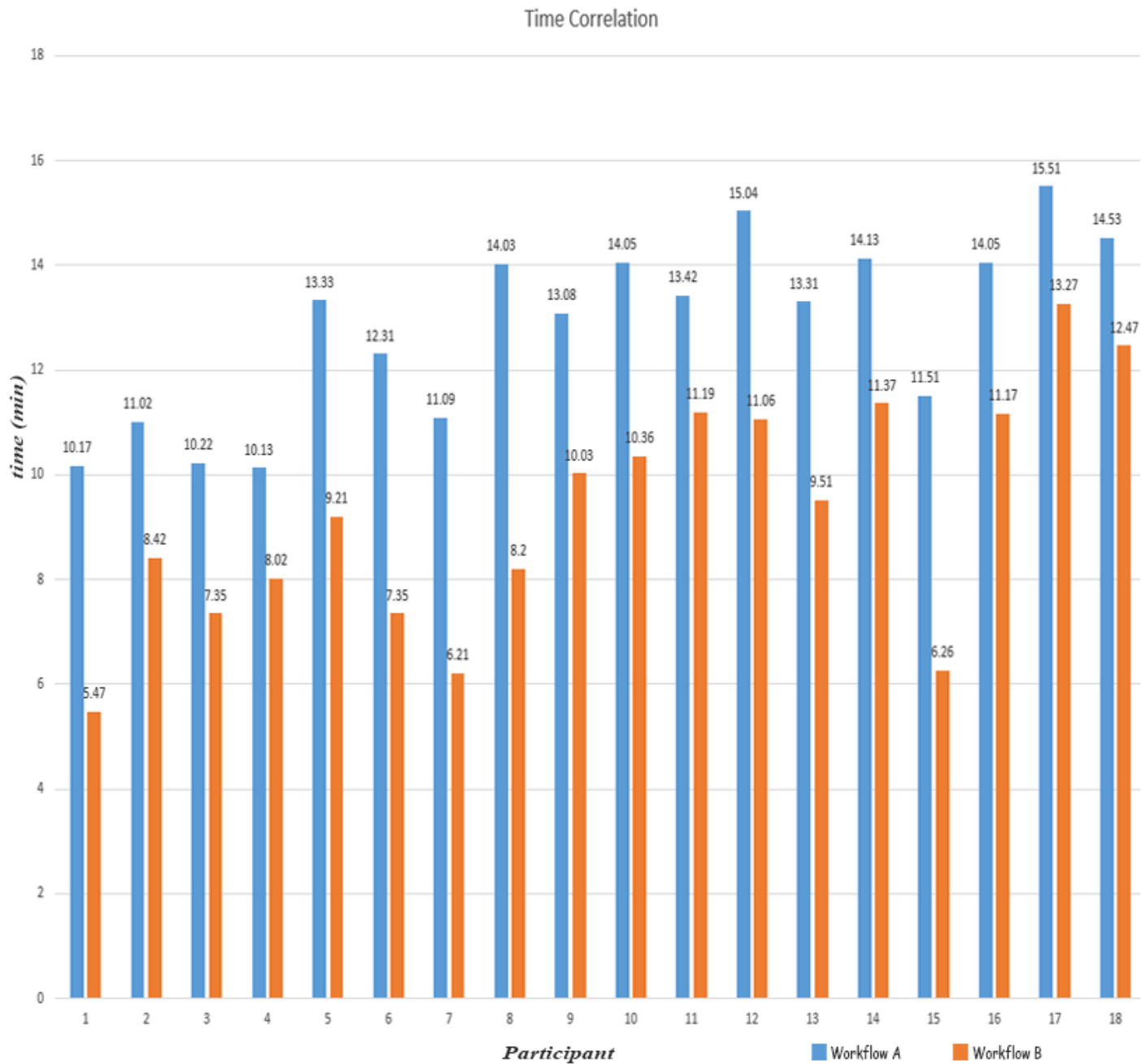
Reducing delay time for the transference of pixels could be accomplished by transferring segments of pixels instead of transferring the whole image. This would allow the user to create textures without having to consider the texture size.

Furthermore, a graphic artist might need to texture parts of the UV layout; this is often done by creating masks manually. Workflow B permits the user to select polygons in the 3D modeling software which then automatically highlight the corresponding UV coordinates in image editor. A feature that allows the image manipulation program to automatically construct a mask for the user, using selected polygons, would allow for an even faster and more functional texturing workflow.

Finally, being able to view and change more texture maps such as normal maps and specular maps in the image manipulation program could be a valuable feature as those maps are used in games, especially if they would automatically be transferred to the 3D viewer and applied to the assets material properties.

Appendices

APPENDIX A



Blue parts show the time it took, in minutes, for each participant using workflow A. The orange parts display the same but for workflow B instead.

The horizontal axis displays each participant and the vertical axis represents the time in minutes.

APPENDIX B

INTRODUCTION

B.1 Swedish

Tack för att deltar i dagens forskning. Du kommer att delta i ett realtid texturerings experiment. Detta experiment är för min kandidatuppsats. Jag undersöker om en annan textureringsarbetsflöde är snabbare än det traditionella sättet att texturera tillgångar för spel.

Vad jag vill att du ska fokusera på när du utvärderar arbetsflödet är vilken av arbetsflöden du föredrar och om att ha flera fönster öppna samtidigt är en distraktion.

Du kommer att bli ombedd att testa två olika arbetsflöden, en som representerar en traditionell metod för texturering och visning (arbetsflöde A) och en som är ganska lik, men som använder realtidsteknik för att visa tillgången (arbetsflöde B).

Experimentet består av flera steg. Varje steg har en sub-uppgift som måste vara klar för att experimentet ska bli fullständig. Jag kommer att ta tid medan du gör experimentet på grund av att kunna jämföra resultaten av dessa arbetsflöden.

På följande skärmar, kommer du först se ett pilottest som kommer att förbereda dig för det verkliga experimentet. Detta innebär att du kan få hjälp ifrån mig eftersom pilottestet inte kommer att mätas på något sätt.

För att jämföra prestanda arbetsflöden i relation till varandra, måste du utföra samma uppgift med både arbetsflöden, och ägna samma mängd arbete när du texturerar de delar/polygoner i båda arbetsflöden. Vad jag menar med det är, om du är strikt vid målningen och vill t.ex. måla kanter, se till att du gör samma sak när du texturear på andra arbetsflödet också.

Efter pilottestet kommer du i princip göra exakt samma sak men den här gången med en svårare uppgift som kommer att mätas med tid.

Efter att ha testat de arbetsflöden, kommer du att bli ombedd att fylla i ett frågeformulär. Om du har någon form av fråga eller problem, tveka inte att be om hjälp.

B.2 English

Thank you for agreeing to participate in today's research. You are about to participate in a real time-texturing experiment. This experiment is for my bachelor thesis. I am researching whether a different texturing workflow is faster than the traditional way of texturing assets for games.

What I want you to focus on when evaluating the workflow is which of the workflows you would prefer and if having multiple windows open simultaneously is a distraction.

You will be asked to test two different workflows, one that represents a traditional method of texturing and viewing (workflow A) and one that is pretty similar but that uses real-time technology to view the asset (workflow B).

The experiment consists of multiple stages. Each stage has a sub-task that needs to be finished in order for the experiment to become complete. I will take time whilst you are doing the experiment due to being able to compare the performance of these workflows.

On the following screens, you will first see a pilot test section that will prepare you for the actual experiment. This means that you can get help from me since the pilot test will not be measured in any way.

In order to compare the performance of the workflows in correlation to one another, I need you to perform the same task with both workflows, and dedicate the same amount of work with texturing the parts/polygons in both workflows. What I mean by that is, if you are strict about painting, for example the borders, make sure that you do the same thing when you are texturing, using the other workflow.

After the pilot test you will basically do the exact same thing but this time with a slightly harder task that will be measured with time.

After testing the workflows, you will be asked to fill out the questionnaires. If you have any kind of questions or issues, please do not hesitate to ask for assistance.

APPENDIX C

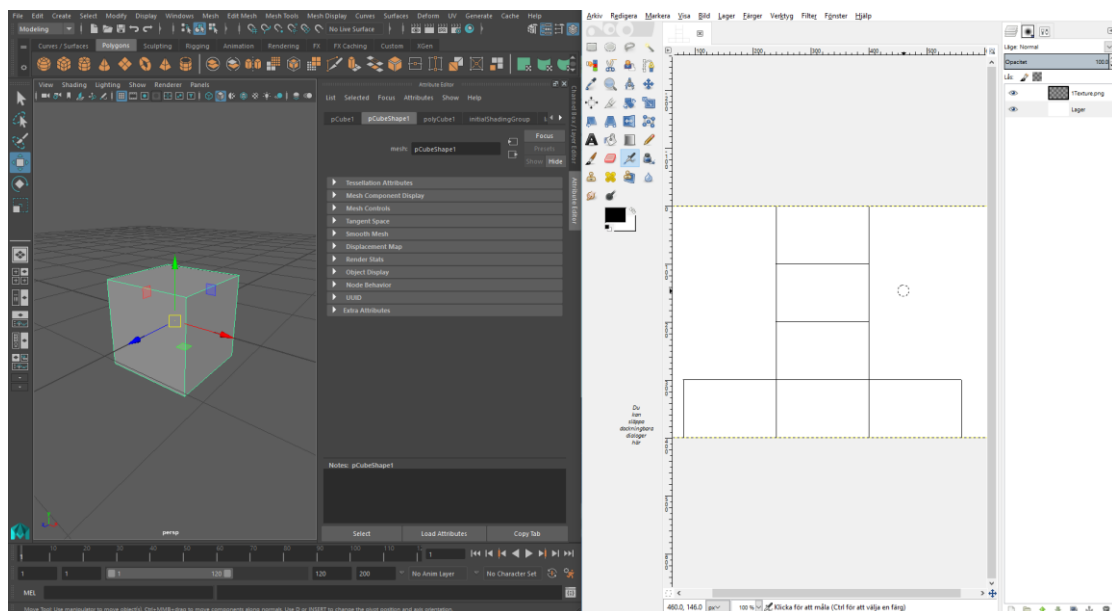
PILOT TEST

In this task you will texture a cube. The layer that you select will be displayed in the 3D viewer.

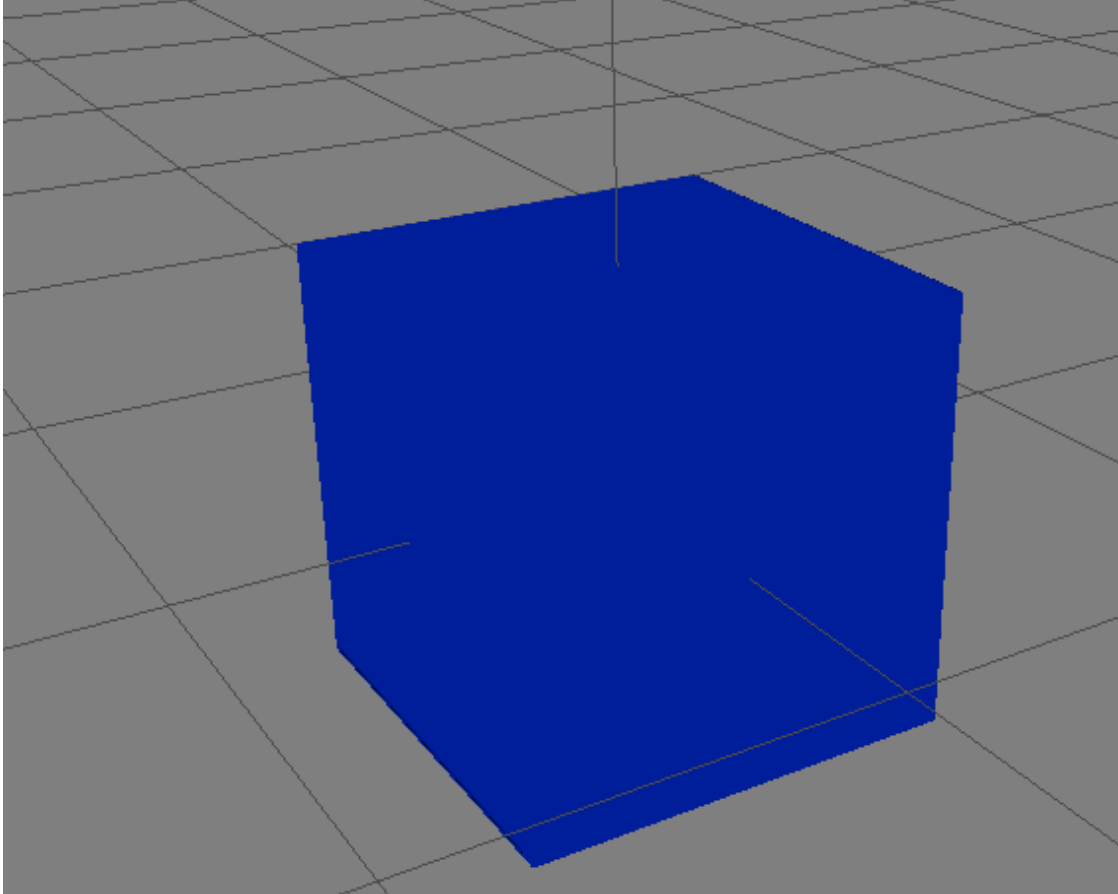
Model: Cube

Steps in Image Manipulation Program (GIMP): 4

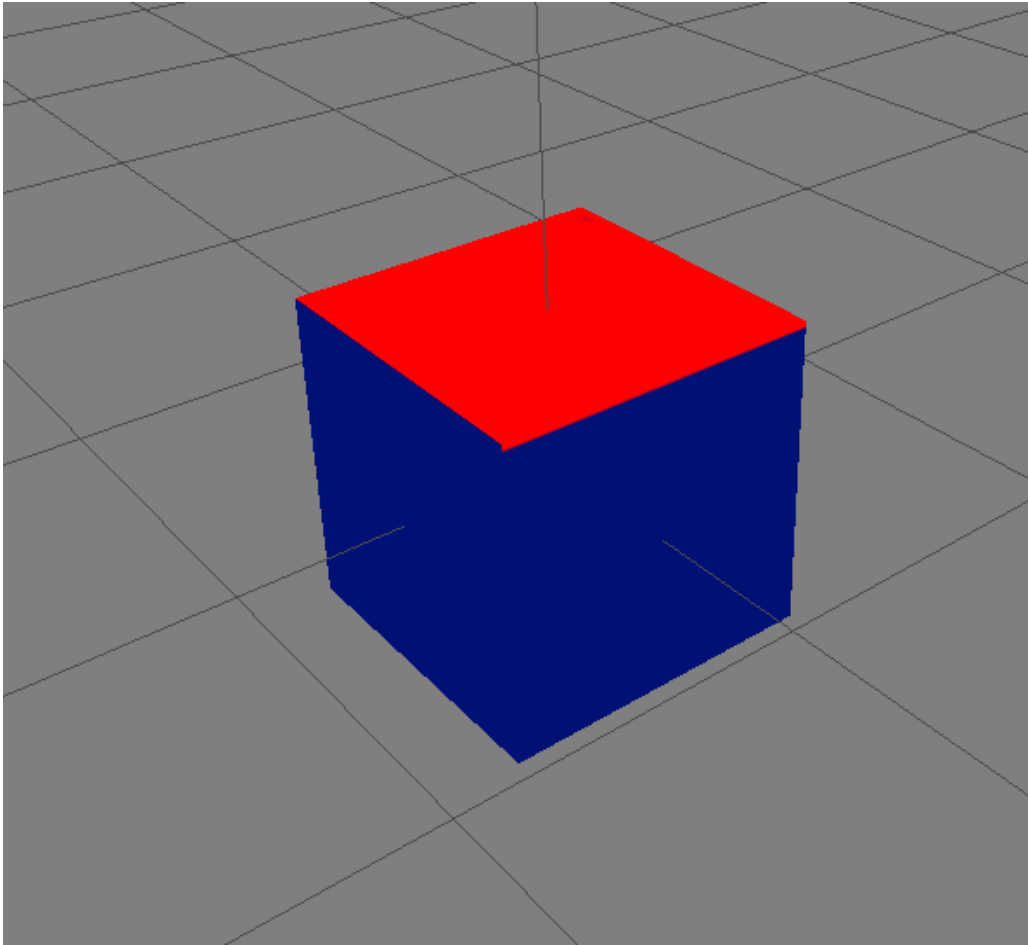
a) Texture the cube using traditional workflow



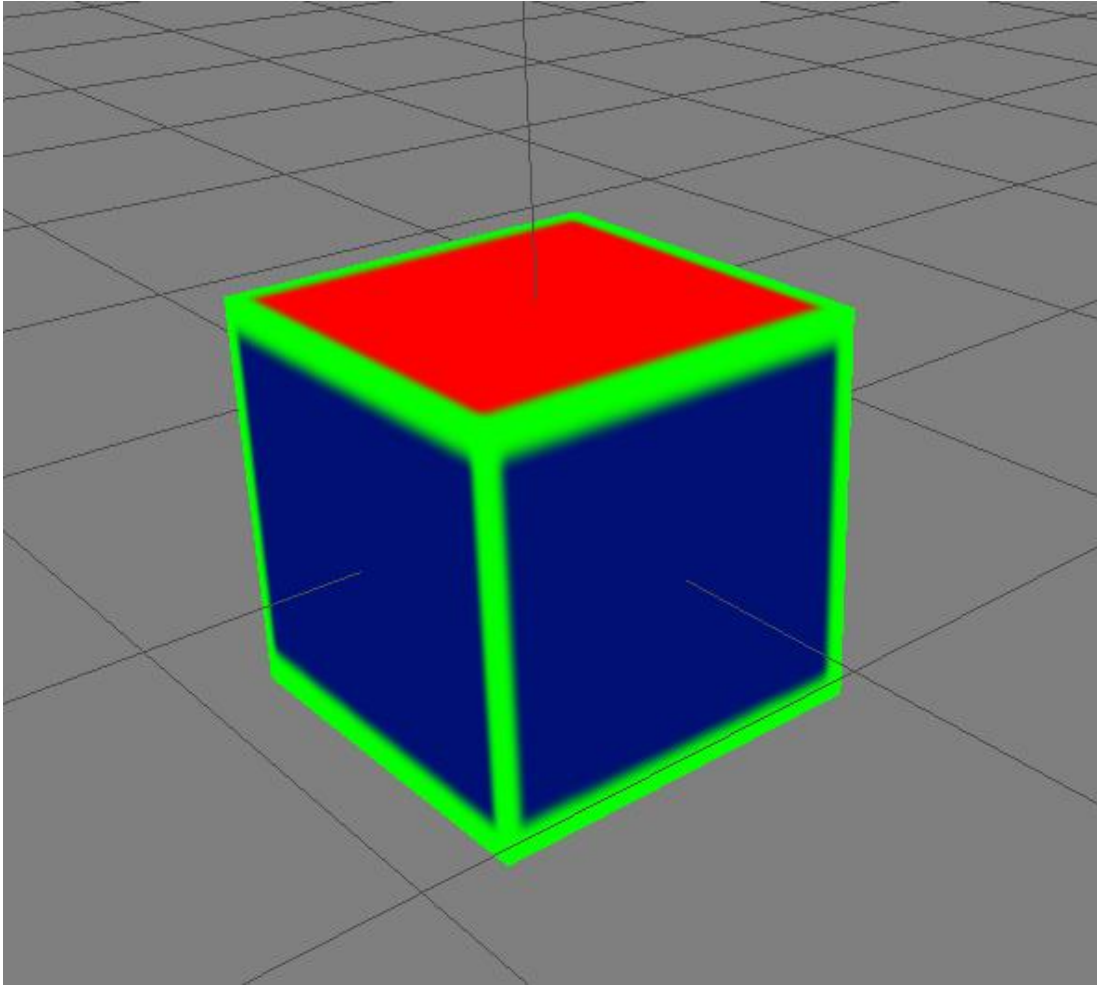
Start by painting the whole cube with blue paint



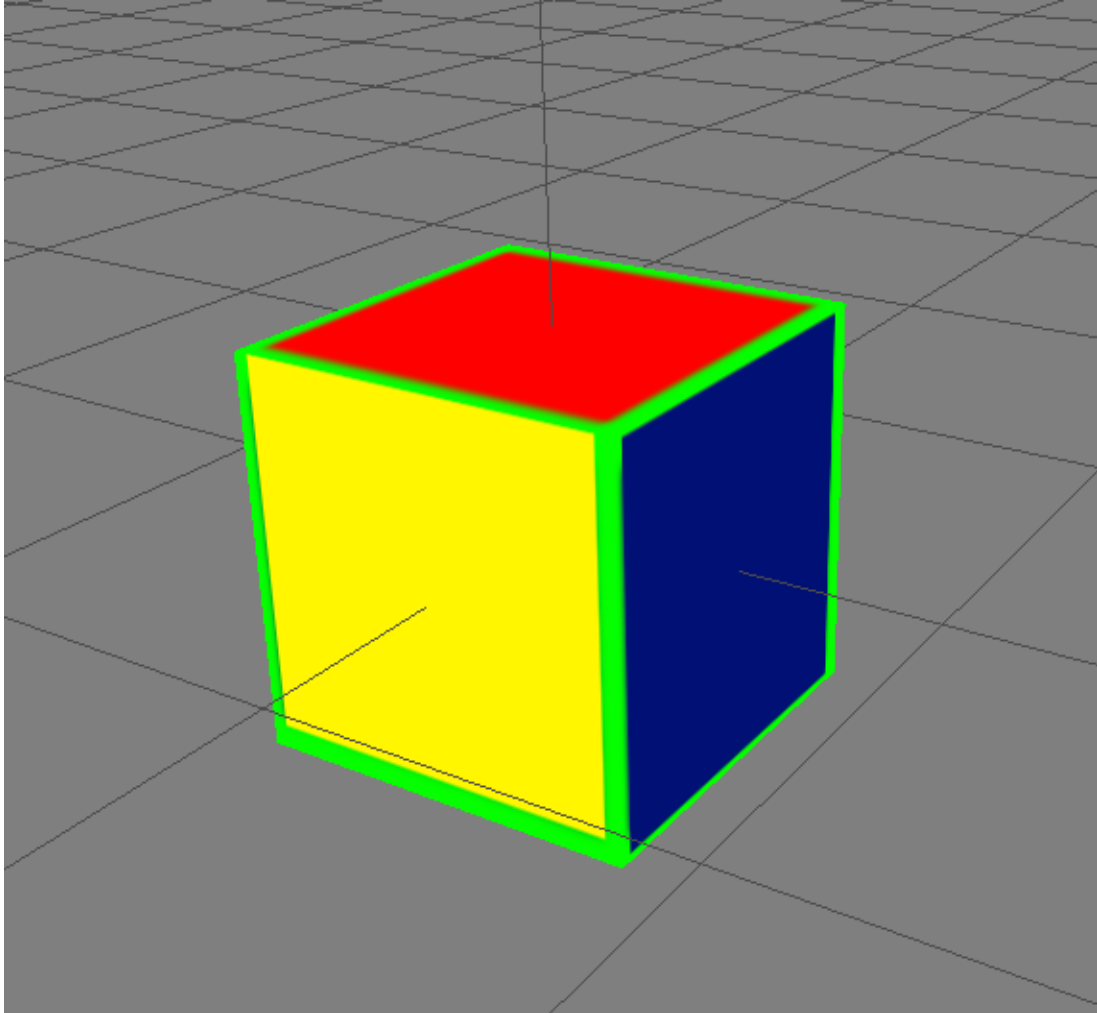
Colour the top polygon with red paint



Colour all the borders with green paint



Colour 4 out of six sides with yellow paint as in the image below (3 of the yellow sides cannot be seen)



b) Texture the cube again, but this time using improved workflow

Well done. You have now finished the pilot test. Let us start with the other texturing task. The clock is about to start ticking...

APPENDIX D

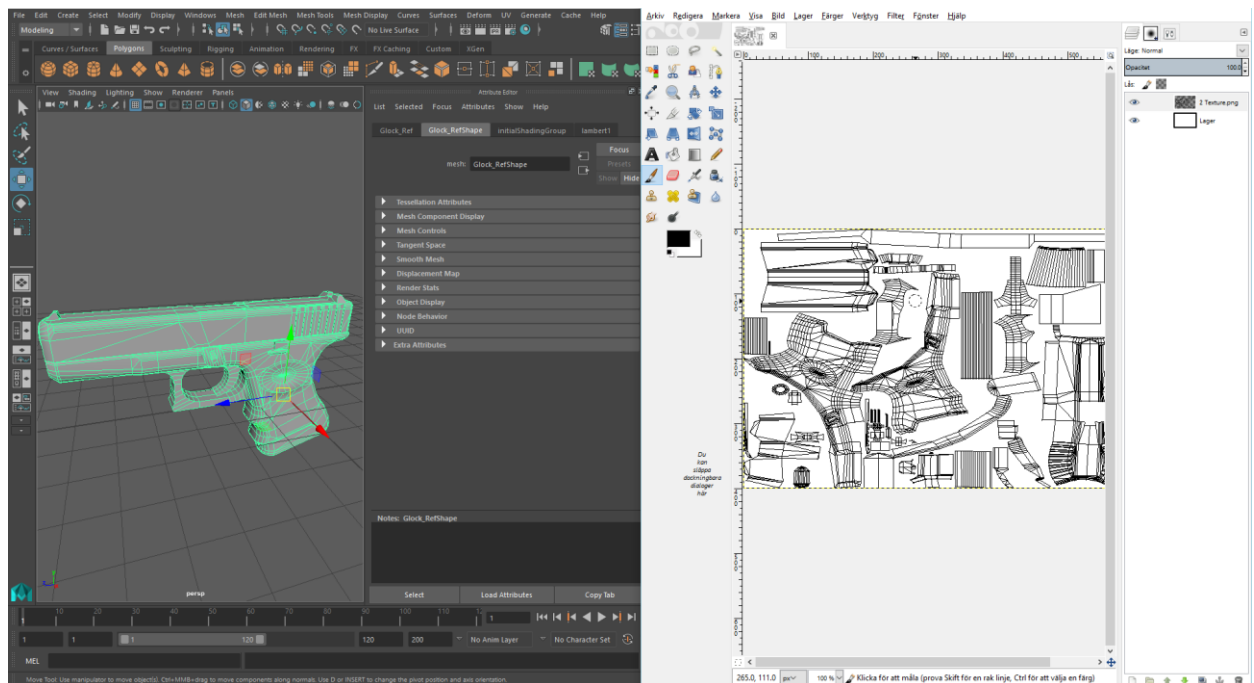
TEXTURING TASK

COUNTER-STRIKE GLOBAL OFFENSIVE SKIN

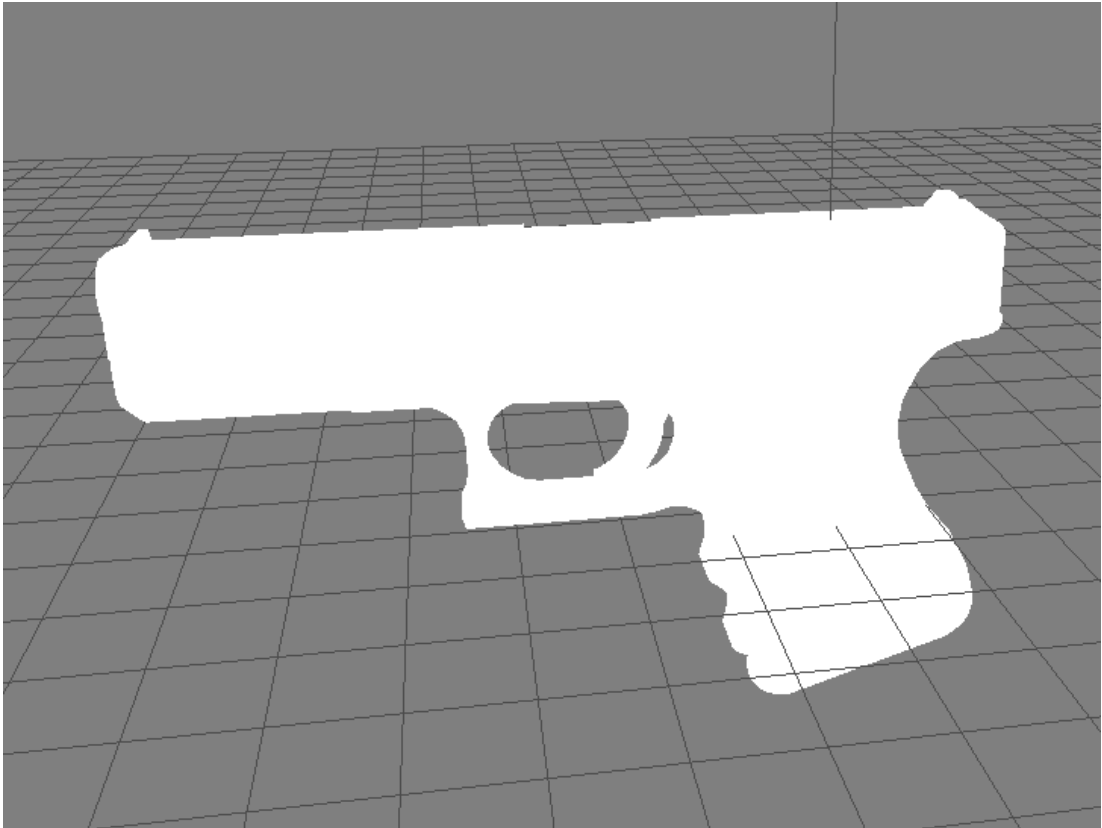
Model: Glock

Steps in Image Manipulation Program (GIMP): 5

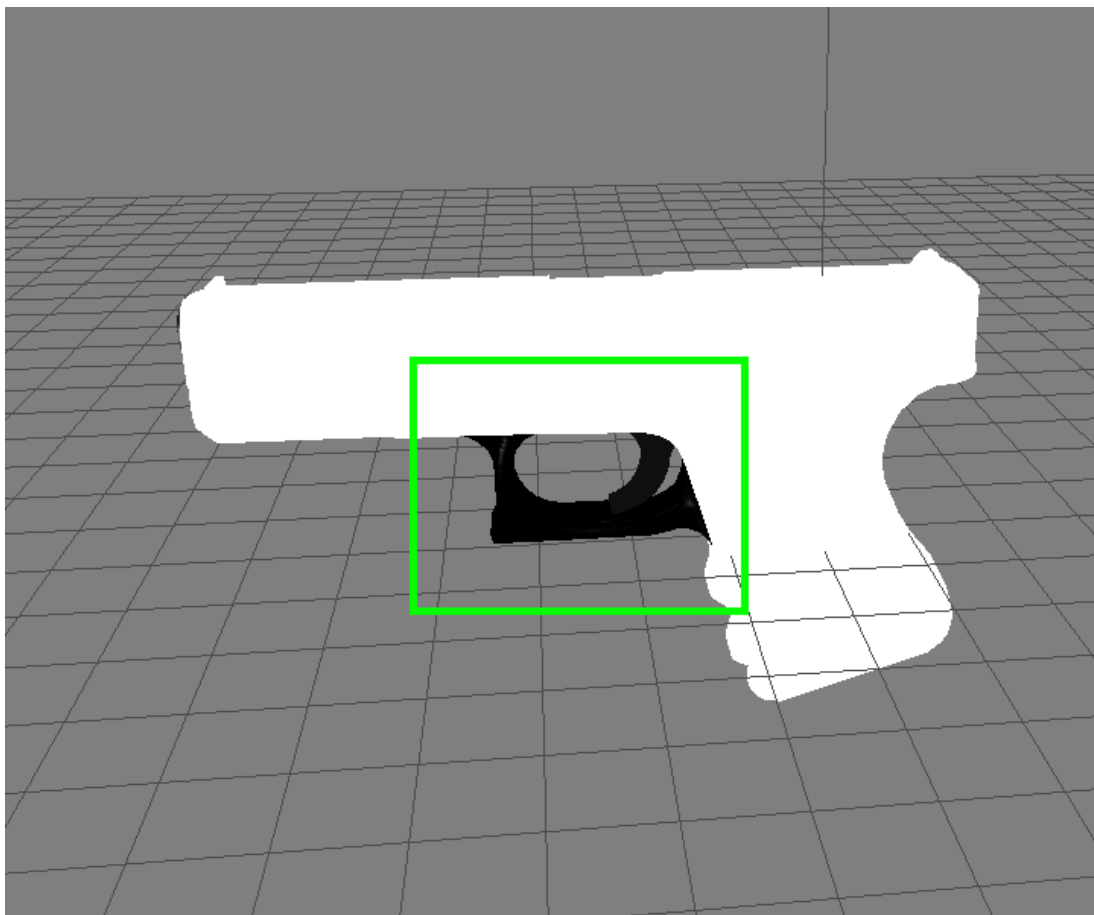
a) Texture the weapon using traditional workflow



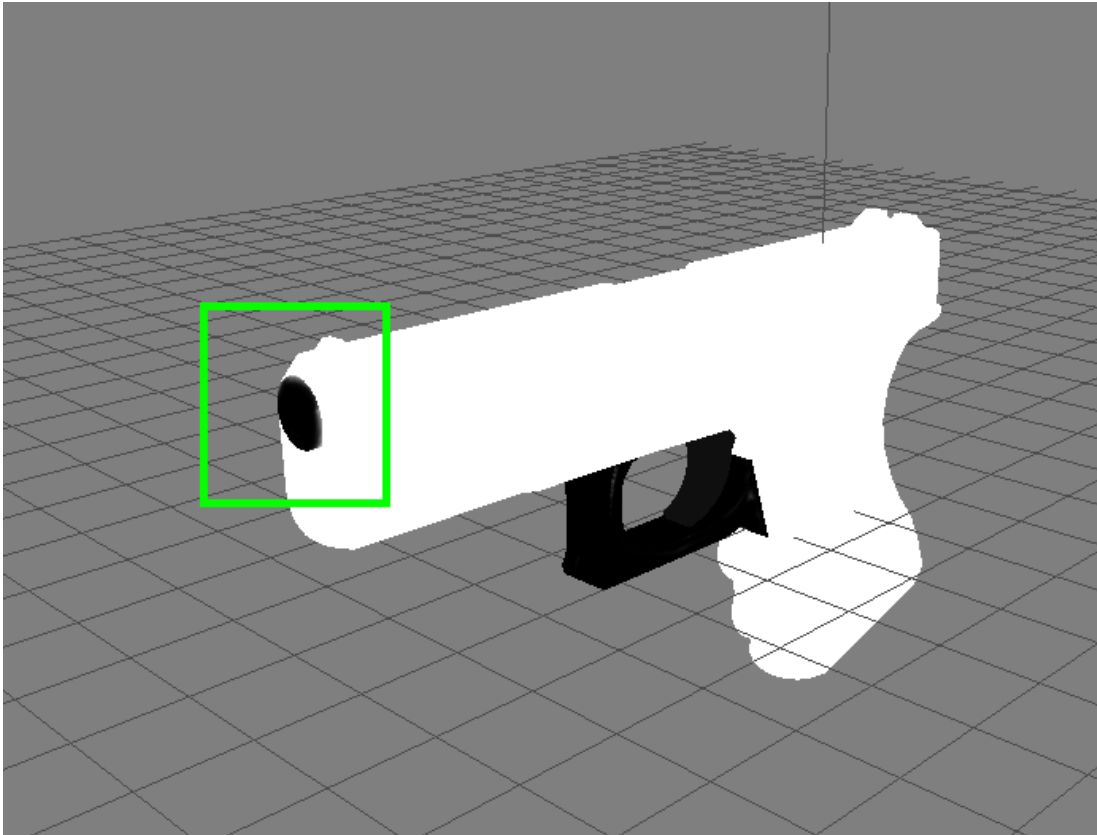
Colour the entire weapon with white paint



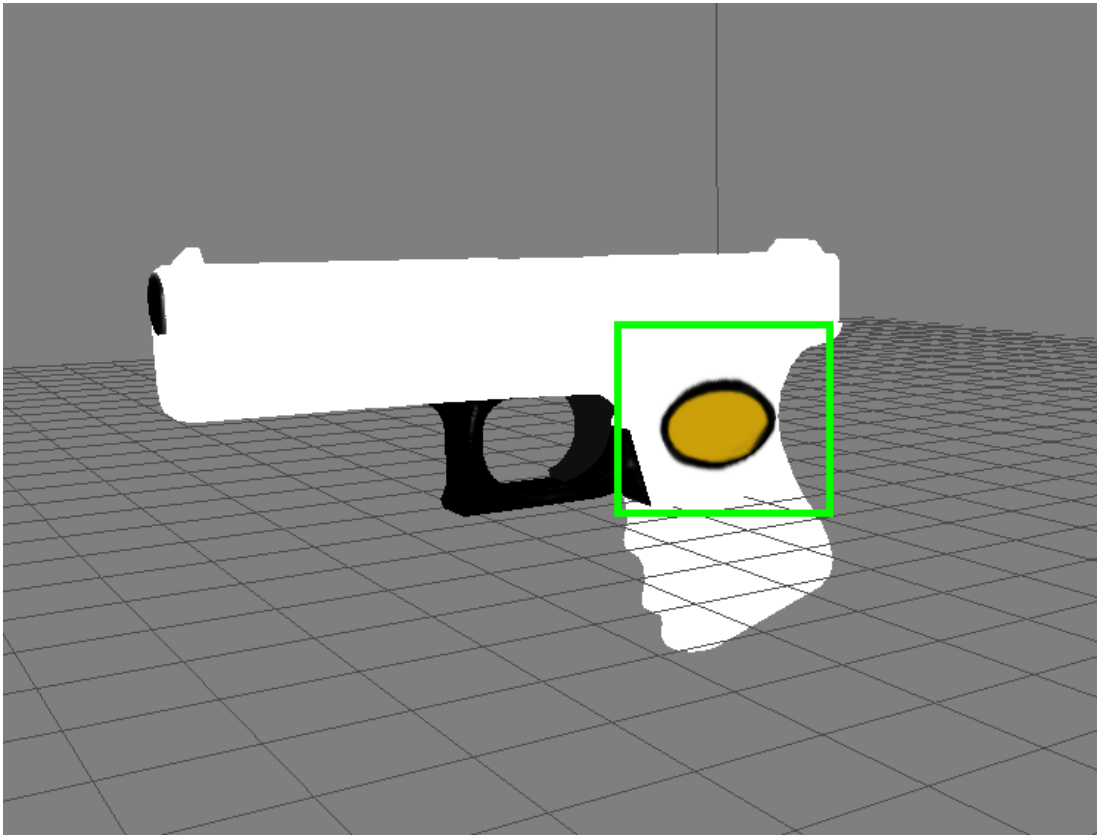
Colour the trigger and the trigger guard with black paint



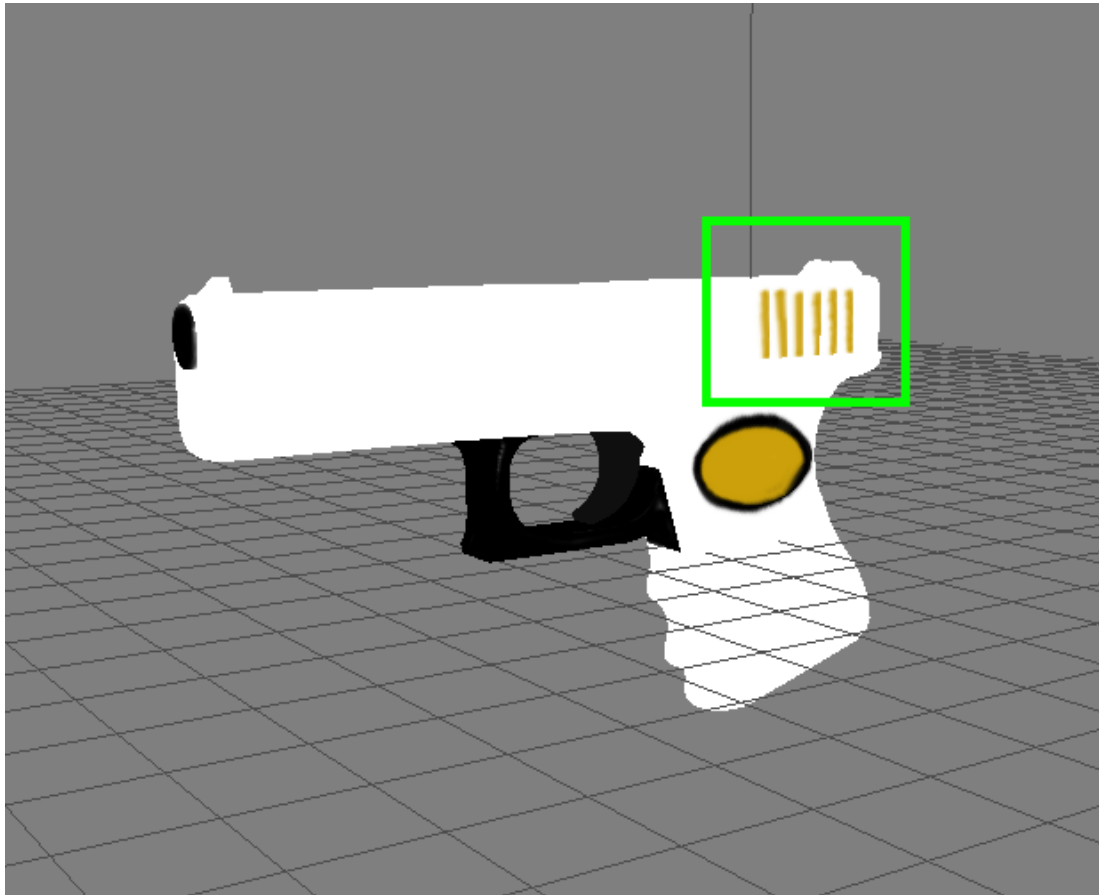
Colour the muzzle/barrel with black paint



Paint the sticker with orange colour, with a black outline



Colour the frame with orange lines. See the figure below!



RESULT



b) Texture the weapon again, but this time using improved workflow

REFERENCES

- [1]. M. Omernick, *Creating the Art of the Game*. New Riders, 2004.
- [2]. Blinn, J. F., & Newell, M. E. (1976). Texture and reflection in computer generated images. *Communications of the ACM*, 19(10), 542-547.
- [3]. L. Lanier, *Advanced Maya Texturing and Lighting (2nd Edition)*. Hoboken, NJ, USA: Wiley, 2008.
- [4]. A. Watkins, *Getting Started in 3D with Maya*. Focal Press, 2012.
- [5]. "File Mapping (Windows)." [Online]. Available: [https://msdn.microsoft.com/en-us/library/windows/desktop/aa366556\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/aa366556(v=vs.85).aspx). [Accessed: 10-May-2016].
- [6]. S. Prahalad and S. Chogle, "SYSTEM, METHOD, AND COMPUTER PROGRAM PRODUCT FOR OPTIMIZING A THREE-DIMENSIONAL TEXTURE WORKFLOW," U.S Patent 20150015574, 15-Jan-2015.
- [7]. "Doc:2.4/Manual/Textures/Painting - BlenderWiki". [Online]. Available at: <https://wiki.blender.org/index.php/Doc:2.4/Manual/Textures/Painting>. [Accessed: 15-mar-2016].
- [8]. "Usability 101: Introduction to Usability". [Online]. Available at: <https://www.nngroup.com/articles/usability-101-introduction-to-usability/>. [Accessed: 15-mar-2016].
- [9]. "ISO 9241-11:1998(en), Ergonomic requirements for office work with visual display terminals (VDTs) — Part 11: Guidance on usability". [Online]. Available at: <https://www.iso.org/obp/ui/#iso:std:iso:9241:-11:ed-1:v1:en>. [Accessed: 20-apr-2016].
- [10]. P. W. Jordan, B. Thomas, I. L. McClelland, and B. Weerdmeester, *Usability Evaluation In Industry*. CRC Press, 1996.
- [11]. Catmull, E. (1974). A subdivision algorithm for computer display of curved surfaces.
- [12]. Paul S. Heckbert. Fundamentals of texture mapping and image warping. University of California, 1989.
- [13]. T. Igarashi and D. Cosgrove, "Adaptive Unwrapping for Interactive Texture Painting," in *Proceedings of the 2001 Symposium on Interactive 3D Graphics*, New York, NY, USA, 2001, pp. 209–216.

- [14]. "Interprocess Communications (Windows)." [Online]. Available: [https://msdn.microsoft.com/en-us/library/windows/desktop/aa365574\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/aa365574(v=vs.85).aspx). [Accessed: 08-Jun-2016].
- [15]. A. P. Sistla, "Distributed Algorithms for Ensuring Fair Interprocess Communicatons," in *Proceedings of the Third Annual ACM Symposium on Principles of Distributed Computing*, New York, NY, USA, 1984, pp. 266–277.
- [16]. "Using Named Objects (Windows)." [Online]. Available: [https://msdn.microsoft.com/en-us/library/windows/desktop/ms686931\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/ms686931(v=vs.85).aspx). [Accessed: 09-Jun-2016].
- [17]. "CS:GO Stash - Browse all skins, stickers and items." [Online]. Available: <https://csgostash.com/>. [Accessed: 09-Jun-2016].
- [18]. "Counter Strike: Global Offensive CSGO Skins." [Online]. Available: <https://www.epicnpc.com/forums/601-Counter-Strike-Global-Offensive-CSGO-Skins>. [Accessed: 09-Jun-2016].
- [19]. "CS:GO Weapon Finish Contest," *polycount*. [Online]. Available: <http://polycount.com/discussion/168659/cs-go-weapon-finish-contest>. [Accessed: 09-Jun-2016].
- [20]. Armstrong. R. (1987). The midpoint on a five-point Likert-type scale. *Perceptual and Motor Skills*.
- [21]. Likert, R. (1932). A technique for the measurement of attitudes. *Archives of Psychology*, 22(140).
- [22]. Dr Nigel J. Miller, University of York. Questionnaires | The Handbook for Economics Lectures.
- [23]. W. Albert and T. Tullis, *Measuring the User Experience: Collecting, Analyzing, and Presenting Usability Metrics*. Newnes, 2013.
- [24]. Douglas W. (2011) *Experimental Design: From User Studies to Psychophysics*.
- [25]. Losby. J. Using Likert Scales in Evaluation Survey Work.
- [26]. Allen. E. and Christopher A. (2007), *Likert Scales and Data Analyses*.