Bilkent University

Department of Computer Engineering

Senior Design Project

Project Specifications Report

Deepgame

Students

Betül Reyhan Uyanık
Mert Alp Taytak
Ömer Faruk Geredeli

Supervisor

Dr. Uğur Güdükbay

Jury Members

Prof. Dr. Özgür Ulusoy
Asst. Prof. Shervin Rahimzadeh Arashloo

Innovation Expert

Cem Çimenbiçer

February 24, 2020

This report is submitted to the Department of Computer Engineering of Bilkent University in partial fulfillment of the requirements of the Senior Design Project course CS491/2.

Table of Contents

1 Introduction

Modern times offer people various kinds of video games through a multitude of platforms. In some genres of games the user plays as a character. Since the appearance of the player character is important be it for immersion or role playing purposes, games have been offering character customization options. Some of these customization options take the form of accessories or clothing that changes the player character superficially. Another option is customization of character models. Some games, with the help of improving technology, offer extensive and elaborate character creation menus. Which makes it possible to create a visually diverse range of characters. However, customization is not limited to visuals. Character voices are another part of the customization process. Due to its nature, human speech is not fit for modeling and customization in the way the human body is. Many games we have seen in the market use a limited set of recorded, generic voice lines for characters. Some do not voice act player character's lines at all. Perhaps, due to how expensive it gets to offer various voices for extensive dialogue options.

We would like to develop a software tool that can be integrated into games themselves to allow the user to put themselves, or maybe others, into the game visually and auditorily. While there exists games with similar features, they require expensive hardware. Using techniques developed in the recent years, our product will work with photographs and audio recordings. Easily acquirable through hardware common to almost every computer user.

1.1 Description

Recent years saw the development of what are commonly referred to as deepfake algorithms. These algorithms allow the user to plant one person's face onto another in a convincing manner. Effectively giving the ability to create images or videos of anyone doing anything. In a related branch, same can be done for the voice and speech, giving the ability to make anyone say anything. However, application of these algorithms to the field of game development is largely unexplored. We propose that these algorithms can be leveraged in game development to allow the player's to put themselves into the game. Moreover, instead of focusing on making numerous models of high quality, a game developer can work on a very high quality generic model instead. High quality facial animations and voice acting can be leveraged by our project to create a highly personalizable game for relatively low cost.

Our product will leverage deepfake algorithms modified for our use case for a software tool that can be integrated into games in order to make in-game models appear like real persons and lines sound like they are in chosen persons' voices. Also, it will be working with images and recordings that can be acquired through everyday hardware.

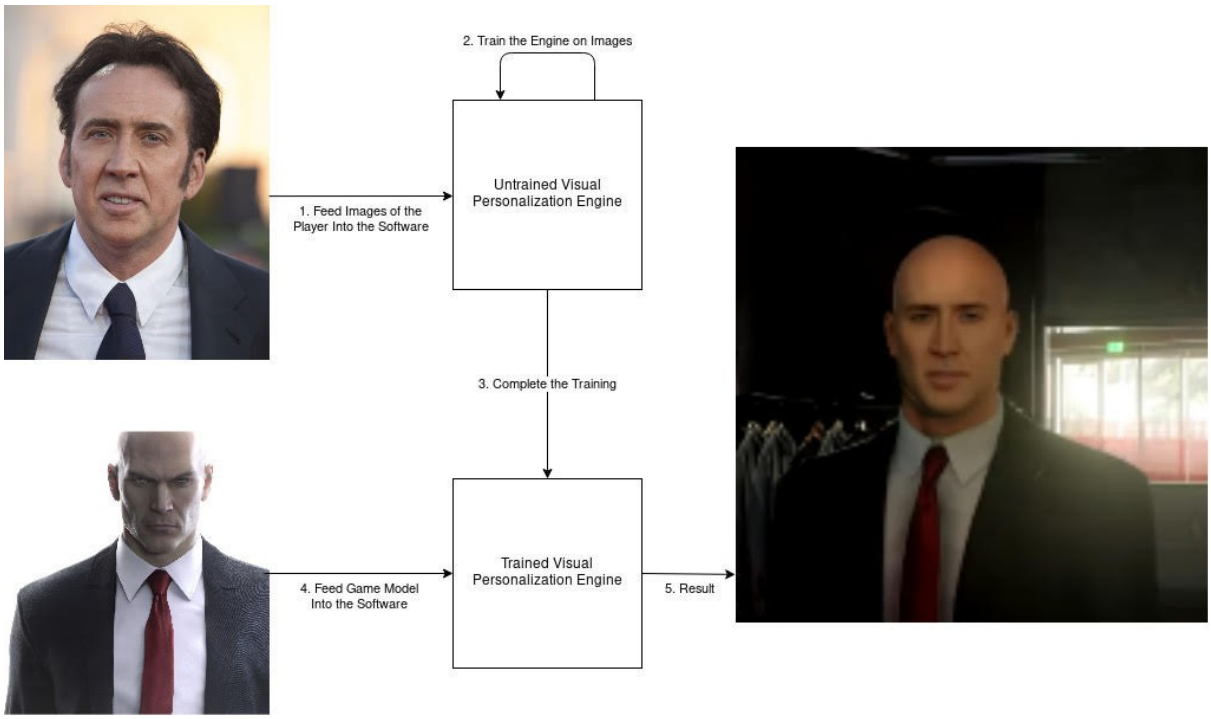

Figure 1. Example Application for the Visual Case

The figure above shows the envisioned application of our product for the visual feature transfer. Games integrating our product will have the option to customize characters via feeding images and recordings to tool's personalization engine. Afterwards the engine will transfer the input features onto the generic model inside where it can be used in the game. A similar approach will be taken with regards to audio, using recordings to make generic lines sound like spoken by the input person.

Since this is a largely unexplored field, we will have to develop our own technology. Therefore, it is hard to make exact specifications without knowing the limitations first. However, in the final product, we would like to make a performant and convincing customization tool in which, possibly, people can create and share packages of their likeness and multiple game characters can be customized through this tool.

1.2 Similar Products and Technologies

Under this section we will discuss products and technologies related to our project. These can be divided into two subjects: Deepfake and Kinect.

1.2.1 Deepfake Technology

Deepfakes are synthetic media in which a person in an existing image or video is replaced with someone else's likeness [1]. While the act of faking content is a not new, deepfakes leverage powerful techniques from machine learning and artificial intelligence to manipulate or generate visual and audio content with a high potential to deceive. The main machine learning methods used to create deepfakes are based on deep learning and involve training generative neural network architectures, such as autoencoders or generative adversarial networks (GANs). Deepfakes rely on a type of neural network called an autoencoder. These consist of an encoder, which reduces an image to a lower dimensional latent space, and a decoder, which reconstructs the image from the latent representation. Deepfakes utilize this

architecture by having a universal encoder which encodes a person into the latent space. The latent representation contains key features about their facial features and body posture. This can then be decoded with a model trained specifically for the target. This means the target's detailed information will be superimposed on the underlying facial and body features of the original video, represented in the latent space.

Following are some applications using the deepfake technology [2].

1.2.1.1 Deepfakes Web ß

This application is a web-based application. This application, which uses the machine learning technology to load the differences in facial data in the visual field, takes an average of 4.5 hours. Although it uses the most powerful GPU on the cloud, it can take hours to process all the data.

1.2.1.2 MachineTube

This application is also a web-based application. Since it does not use cloud technology, the speed of the application is based on the power of your computer. This app also needs high graphic features. The processing time of this application takes hours.

1.2.1.3 DeepFaceLab

This is a Windows application. This application is used in research and informatics students. Since it does not have a user-friendly interface, it takes time to learn, but it is one of the most successful applications.

1.2.2 Kinect Technology

It is worth discussing briefly about Kinect technology. Kinect technology is the technology of playing games with only hand, arm and leg movements without using peripherals (weapons, console tools) in computer environment [3]. In addition, sound is valid for a separate communication. This technology is realized with red infrared projectors by sensing hand, arm and leg movements. It converts the data obtained by these rays into command with CMOS sensors and thus, it is perceived as a command. Kinect is only responsible for delivering commands.

Following is the technology used in a XBOX game that uses the Kinect camera as a 3D sensor to model the player’s face and body.

1.2.2.1 XBox Kinect Sports Rivals ‘Champion’ Scan

A new generation feature that makes scanning your face and body possible only with Kinect to create a digital, stylized version that will represent you in the game. After combing your body, Kinect focuses on your face and captures 93 different measurements to create a 3D model. Then it adds features like skin color and hair color. Thus, it creates unique game characters [4].

1.3 Constraints

1.3.1 Implementation Constraints

- Git and GitHub will be used for version control and issue tracking.

- An existing game or game engine will be modified to implement our project.
- Our software will work in local, without cloud usage.
- Modified deepfake algorithms will be used for feature transfer.
- Our software will target PC as the platform.

1.3.2 Economic Constraints

- No additional hardware, other than those required to play high end games, will be necessary to use the product.
- Only free, open source libraries and tools will be used.

1.3.3 Ethical Constraints

- We will act with respect to protecting people's right to their likeness.
- We will create a license so that people's biometric data can be shared only with their consent.

1.3.4 Security Constraints

- Any biometric data will be encrypted.
- Input images and recordings will not be kept beyond training.

1.3.5 Time Constraints

- Project deadlines will be met.

1.4 Professional and Ethical Issues

Prime ethical concern is people's right to their likeness. There is already controversy surrounding unauthorized use of people's likeness in deepfake algorithms to create unethical videos. In order to present such unethical uses, we will establish a license that gives everyone legal rights to their likeness. Images of people can be easily acquired through social media and what people do in private cannot be stopped. However, we hope that giving the people legal rights will be enough to stop public sharing of anything unethical using our product.

Another issue is that the games using our product will need images and recordings of users. There is a privacy risk that those data can be stolen. In order to prevent this we will make our tool such that it erases any data after training and works in local.

2 Requirements

2.1 Functional Requirements

2.1.1 Visual Feature Transfer

- User is able to create an in game character of their likeness by feeding our software images of themselves.
- This character is a convincing model of the user and can be played as any other character in the game with further customization options such as accessories.
- As an advanced feature, videos can be processed to extract mimics as animations.

2.1.2 Auditory Feature Transfer

- User is able to customize a voice package using recordings of their voice.

2.1.3 Feature Packages

- Results of visual and auditory feature transfers can be packaged and shared in a secure and private manner.

2.2 Non-functional Requirements

2.2.1 Performance

- Training process for each feature should take less than an hour.
- Resulting models should be usable in real time with no major performance cost.
- Input lag should be less than 50 milliseconds.

2.2.2 Achieved Similarity

- Resulting models should appear and sound reasonably convincing.

2.2.3 Privacy

- User data should only be accessible and usable through their consent.
- Malicious use of the product that can undermine people’s privacy should be prevented.

2.2.4 Security

- Any sensitive data should be encrypted if permanence is necessary, deleted after use if not.
- Only the user owned hardware should access user data during operation, i.e. no cloud processing.

3 References

[1] A. Zucconi, “Understanding the Technology Behind DeepFakes,” Alan Zucconi, 18-Apr-2018. [Online]. Available: https://www.alanzucconi.com/2018/03/14/understanding-the-technology-behind-deepfakes/. [Accessed: 23-Feb-2020].

[2] B. Staff, “8 Best Deepfake Apps and Websites You Can Try for Fun,” Beebom, 06-Sep-2019. [Online]. Available: https://beebom.com/best-deepfake-apps-websites/. [Accessed: 23-Feb-2020].

[3] “Azure Kinect DK – Develop AI Models: Microsoft Azure,” – Develop AI Models | Microsoft Azure. [Online]. Available: https://azure.microsoft.com/en-us/services/kinect-dk/. [Accessed: 23-Feb-2020].

[4] X. W. Staff, “Creating a Champion in Kinect Sports Rivals,” Xbox Wire, 20-Oct-2015. [Online]. Available: https://news.xbox.com/en-us/2014/04/01/games-ksr-developer-qa/. [Accessed: 23-Feb-2020].