

Mimicking Facial Expression from Actor to Virtual Avatar using Machine Learning

Daniel Woldegiorgis



Thesis submitted for the degree of
Master in Cybernetics and Autonomous Systems
30 credits

Department of Informatics
The Faculty of Mathematics and Natural Sciences

UNIVERSITY OF OSLO

Autumn 2021

Mimicking Facial Expression from Actor to Virtual Avatar using Machine Learning

Daniel Woldegiorgis

© 2021 Daniel Woldegiorgis

Mimicking Facial Expression from Actor to Virtual Avatar using Machine Learning

<http://www.duo.uio.no/>

Printed: Representralen, University of Oslo

Abstract

Training police officers to interview abused children is a challenging and time-consuming undertaking. Traditionally, a child actor has been hired to help as the interview object. Due to resource constraints and the time required to teach the child actor, this is not easily scalable. Unfortunately, interviewing abused children is a challenging task, as the majority of sexually abused children exhibit no visible signs of assault [3]. Machine learning enables us to make the essential training schedule more accessible.

With the advancements in machine learning, we may be able to design a training regime aided by machine learning that can replace the child actor with an interactive photo-realistic child avatar capable of meaningful interaction with the trainees. This thesis focuses on designing a method for the visualization of the avatar, providing a photo-realistic appearance to the avatar in which we are cable of controlling the facial movement while also taking emotion into account, resulting in a real-time re-enactment of a child's facial expressions. Our approach is based on a modified method from [35] for synthesizing the lower facial texture for the avatar and mimicking facial expression from a source actor to drive the child avatar.

Acknowledgments

I would not have been able to complete my thesis without the assistance of various folks along the way. To begin, I'd want to show my gratitude to Michael Riegler and Pål Halvorsen, who served as my supervisors. Additionally, I'd like to thank Carl Bendiks for providing me with a data set, and I'd want to thank my fellow classmates and friends for their encouragement and support during this trying semester. Your encouragement helped keep me going!

Contents

Abstract	1
Acknowledgments	2
1 Introduction	1
1.1 Motivation	1
1.2 Problem Statement	2
1.3 Scope and limitations	4
1.4 Research methods	4
1.5 Main contribution	5
1.6 Thesis outline	6
2 Background	8
2.1 Introduction	8
2.2 Virtual Avatars for Investigative Interviews with Children	8
2.3 Computer vision	10
2.3.1 Digital image	10
2.4 Machine Learning	12
2.4.1 Supervised Learning	12
2.4.2 Unsupervised learning	17
2.4.3 3D Morphable Face Models	18
2.5 Metrics	20
2.5.1 L2 distance	21
2.5.2 Normalized Mean Error	21
2.5.3 Learned Perceptual Image Patch Similarity	21

2.5.4	Structural Similarity Index Measure	22
2.6	Temporal Optical Flow	22
2.7	Temporal Learned Perceptual Image Patch Similarity	23
2.8	Summary	23
3	Methodology	25
3.1	Feature extractor	27
3.2	Facial texture synthesizer	29
3.3	Rendrer	33
3.4	Blender	33
3.5	Smoothing	36
3.6	summary	36
4	Experiments and results	38
4.1	Subjective comparison of models	38
4.1.1	Data	38
4.1.2	Questionnaire	40
4.1.3	Results and Discussions	41
4.2	Quantitative evaluation	44
4.2.1	Quantitative evaluation when source avatar is equal target avatar	47
4.2.2	Evaluation of n best candidates	49
4.3	Limitations	50
4.4	System resources	51
4.5	Summary	52
5	Summary and Conclusions	54
5.1	Main contribution	56
5.2	Future work	56
A	Google Forms Questionnaire	62
A.1	Answers	62
A.2	Questions	65

List of Figures

2.1	multimodal model for interview training	9
2.2	Convolution example	11
2.3	Feed-forward neural network	13
2.4	Convolutional neural network arrangement	15
2.5	Max pooling	17
2.6	3D morphable face model	18
2.7	Basel face model distribution	19
3.1	Model architecture	26
3.2	Extending and cropping BFM	29
3.3	Preprocessing	30
3.4	Gaussian pyramid [2]	34
3.5	Laplacian pyramid [2]	35
3.6	Example blending	36
4.1	Data set used in this case study	39
4.2	Average L2-distance, Obama vs Child	45
4.3	L2-distance, where source identity is equal target identity	46
4.4	SSIM, LPIPS and NME for both avatars, methods and base loss, for both for SSIM and LPIPS lower y-value is better while higher is better for NME	48
4.5	tLP and tOF for both avatars, methods and base loss, for both metrics lower y-value is better	49
4.6	SSIM value for different n best candidate values. Base is when true vertex-colors are known	50
4.7	Limitation example	51

A.1	Video of reenacted Obama avatar using method 1, and questions from question portion 1, see table 4.2. User is asked to score the method.	66
A.2	Video of reenacted Obama avatar using method 2, and questions from question portion 1, see table 4.2. User is asked to score the method regarding given topic.	67
A.3	Side by side comparison of method 1 and 2 for Obama avatar, with the original source actor for reference and a separate video of the original target video. User is asked to describe thoughts about overall feel and quality of reenacted avatars	68
A.4	Video of reenacted child avatar using method 1, and questions from question portion 1, see table 4.2. User is asked to score the method regarding given topics.	69
A.5	Video of reenacted child avatar using method 2, and questions from question portion 1, see table 4.2. User is asked to score the method regarding given topics.	70
A.6	Side by side comparison of method 1 and 2 for child avatar, with the original source actor for reference and a separate video of the original target video. User is asked to describe thoughts about overall feel and quality of reenacted avatars.	71
A.7	Comparison of child and Obama avatar, user is asked to choose the preferred avatar for given questions, see table 4.3. User is also encouraged to give feedback on aspects to improve and state gender and age group	72

List of Tables

4.1	Data set used in testing	39
4.2	Question set 1	41
4.3	Question from third portion	41
4.4	Answer for questions regarding Obama avatar for table 4.2	42
4.5	Answer for questions regarding child avatar for table 4.2	42
4.6	Scores for child vs Obama for questions from table 4.3	43
4.7	Which avatar did the participants prefer and why? Question portion 4.	43
4.8	Feed back from the participants regarding improvement	44
4.9	Time measurement for non-negligible time consuming methods .	52
A.1	Answers to Obama questions, fig A.1 and A.2	62
A.2	Answers to child questions, fig A.1 and A.2	63
A.3	Answer for fig. A.3	63
A.4	Answer for fig. A.6	64

Chapter 1

Introduction

Recent advancements in computational resources and data gathering have contributed to the widespread adoption and application of machine learning. Machine learning is a popular topic, and the popularity does not seem to slow down. The possibility of using machine learning to solve novel problems is widely attractive; this thesis will exactly try to accomplish this.

1.1 Motivation

Conducting investigative interviews with child abuse victims is a complicated and difficult undertaking, as children are generally both victims and primary witnesses. Often, particularly in cases of suspected domestic violence and child sexual abuse, the child is the primary source of information regarding the abusive incidents [6]. The majority of sexually abused children exhibit no physical indications of abuse [3]. This means that the progress of the police investigation is contingent upon the child's account of the incident and the interviewer's ability to get the maximum amount of information from the child in the best possible manner. Given the gravity and complexity of this task, training the interviewer to conduct an effective interview requires a significant amount of resources and training opportunities in realistic interview settings, which are not always available.

Numerous international studies from a variety of countries have shown

that poor quality investigative interviews are a widespread problem [9] [23], sometimes resulting in the invalidation of evidence and the dismissal of cases due to procedural errors. The urgent need for a cost-effective and comprehensive training program is critical.

A recent study indicates that using gaming and avatars to teach interviewers for investigative interviews of child abuse victims may be effective, especially when training is integrated with feedback and provided over a longer time period [26] [30].

However, no system exists at the moment that enables realistic avatar interviewees to react audibly to every question posed by the interviewer. At the moment, interview training requires trained actors, making training prohibitively costly to offer - training an actor may take up to three months, and hourly fees are exorbitant [5].

Recent advancements in artificial intelligence allow the building of a complete interviewer training program incorporating a range of machine learning techniques. A system powered by artificial intelligence might be utilized to create a dynamic interactive training program that uses a realistic avatar to replicate a variety of realistic circumstances with which the conductor can engage. This system may be composed of several orthogonal components that are quite often reduced further; this thesis will concentrate on the generation/synthesizing of photorealistic mouth motion (speech). This task may be solved using a variety of machine learning techniques, we will attempt to approach this in the simplest and most straightforward possible manner, emphasizing achievability rather than performance.

1.2 Problem Statement

The multimodal avatar model proposed in [5] consists of multiple parts that are orthogonal to each other. In this thesis, we will focus on the system related to generating the visualization of the avatar. We intend to use an actor to reenact an avatar; more precisely, we wish to determine whether we can make the visual avatar's facial expression follow the actor audio, which leads to our first thesis

objective;

Objective 1: *Research whether we can make the visual avatar's facial expression follow the audio by reenacting the expression associated with the source actor's audio.*

The problem of facial reenactment is a well-studied subject in machine learning and computer vision, which have received increased attention in recent years as a result of the popularity and development of deep neural networks. Prior to the advent of deep neural networks, a common technique was to produce and manipulate the desired face and appearance using a 3D morphable model (3DMM). Today, a typical technique is to employ a deep neural network in conjunction with or without a 3DMM. Typically, the 3DMM is used to parameterize a face's appearance and shape, which is subsequently used as an input to a deep neural network, as demonstrated in [33][38][39]. Because generation of the 3DMM is a time-consuming procedure that requires numerous face scans in an ideal situation, we will use a pretrained/constructed 3DMM; this is also true in some cases for deep neural networks. Face creation and reenactment is a complex task that typically demands a large data set for a deep neural network to obtain meaningful knowledge. The data set utilized frequently dictates the model's performance and is prone to include a bias due to an imperfect distribution. We will therefore;

Objective 2: *Compare the performance of our model on two distinct avatars and examining the effect of the data set on the models performance.*

We intend to compare an adult avatar to a child avatar; it's natural to suppose that trained models have a bias toward the mean population, and in this case, an adult face is more likely to be more accurately represented by the data set used to train the face models utilized in this thesis. Additionally, restrictions pertaining to children's privacy may result in a limited representation of children in the models used, further resulting to assumably worse performance for the child avatar. This may be a cause for concern, given that one of the thesis's main purpose is to contribute to the development of a child avatar.

1.3 Scope and limitations

Due to the complexity of complete facial reenactment, we will mainly focus on reenacting the lower mouth region. This is accomplished using a modified approach of methods described in the [35]. The paper approaches the task of visual speech synthesizing in a straightforward and exact manner. The paper proposes synthesizing the lower mouth region using a weighted median of similar images; the weighted median maintains temporal coherence while retaining the image's sharpness. The paper computes the weighted median for each pixel coordinate in a stack of n similar images. Due to the high computational cost of this approach, we propose sampling the pixels and computing the weighted median on the sampled pixels. By employing a 3DMM model, we can represent each face in the n stacked images with a 3D mesh and its associated set of vertex colors. We can then compute the weighted median for each corresponding vertex color in the n stacked images, which we utilize to generate the synthesized image.

A questionnaire is used to evaluate the model for two different avatars compared to one another. The questionnaire is organized using Google forms and includes questions about the quality of the reenacted avatars; participants are also asked to share their thoughts on what they thought was good and poor.

1.4 Research methods

Our research method is based on the Association for Computing Machinery (ACM) method "Computing as a Discipline" [13]. The paper proposes a framework for the discipline of computing created by a task force assigned by ACM Education Board, and it describes three paradigms: theory, abstraction, and design.

Theory: The theory paradigm is related to mathematical coherent and valid theory. It includes four stages; (i) characterize objects of study (definition), (ii) hypothesize possible relationships among them (theorem), (iii) determine whether the relationships are true (proof), and (iv) interpret

results.

Abstraction: The abstraction paradigm is rooted in the field of the experimental scientific method and consists of four steps; (i) form a hypothesis, (ii) construct a model and make a prediction, (iii) design an experiment and collect data, (iv) and analyze results

Design: The design paradigm is rooted in engineering. The paradigm consists of four steps; (i) state requirements (ii), state specifications, (iii) design and implement the system, (iv) test the system.

Our research is mostly based on the design paradigm; we began by creating a requirement by outlining the objective of our model and what is necessary for the model to function. Then, we discussed the model specifications such as preprocessing of the model data, which we followed up with the model's design and implementation, as well as final testing.

1.5 Main contribution

The research completed during the thesis can be separated into two major sections: development and evaluation, which correspond to the problem statement assigned in section 1.2. Our main contribution is as follows:

Objective 1: *Research whether we can make the visual avatar's facial expression follow the audio by reenacting the expression associated with the source actor's audio.*

We developed a model capable of reenacting lower facial expressions from an actor to a target avatar. The model parameterizes the actor's expression using a CNN model from [19] and a face detector from [43], which we then fused with the target avatar's identity shape to generate a 3D face mesh using a 3DMM from [29]. Additionally, we customized and applied the facial texture synthesizer from [35] to render the lower face texture onto the 3D face, and then blended the rendered lower face onto the target avatar.

Objective 2: *Comparing the performance of our model on two distinct avatars and examining the effect of the dataset on the model performance.*

We used a questionnaire to assess the model on an adult and child avatar, and examined why the adult avatar was perceived as more realistic by the survey participants than the child avatar. We noticed that when the model actor was more similar to the target avatar, the model performed better. Additionally, we employed actors that matched the target avatars to determine whether the sub-methods (the facial feature extractor and face detector) performed equally well on child and adult actors and found that it overall performed slightly better in the child avatar case.

When the actor's face identity is different from the avatar's, reenacting the child avatar using the actor's facial expression becomes noisy, resulting in the audio not matching the avatar expression. This is because the expression parameter is specified in relation to the face shape/identity; in order for this to work correctly, we must account for the identity mismatch. The facial expression method from [38] can be used to solve this issue, which involves estimating a transformation matrix such that the expressions can be projected to the same space, which makes it simpler to find the child expression that fits the actors audio. Overall this thesis is a small step toward solving the the visual part of the multimudual model (see, fig 2.1). Although the model's quality and performance are insufficient for usage in the final project, the work completed demonstrates how we may control the avatar's emotion using a 3DMM.

Our full model implementation is available on Github¹.

1.6 Thesis outline

The thesis is organized as follows:

Chapter 2 - Background: This chapter provides a basis context for the thesis.

To begin, we provide a concise overview of the main technologies utilized in this thesis, including computer vision and machine learning, more

¹Gitub: <https://github.com/Danielwoldis/masteroppgave>

precisely neural networks. Additionally, we provide a brief overview of the 3D morphable model used in the thesis, and we highlight the metrics employed in chapter 4 (result and experiments).

Chapter 3 - Methodology: This chapter discusses the methodology that was utilized to create the theses. We divide our algorithm into sections and offer a step-by-step explanation of the process from a source image to the reenactment of the target avatar using the source expression. We focus on the face synthesizer method but also provide a good description of the other methods used.

Chapter 4 - Experiments and Results: This chapter summarizes the results of the experiments that were conducted. Additionally, we include a questionnaire comparing the model to two different avatars, an assessment of the system's (model) loss and a brief summary of time measurement, limitations, and finally, a discussion on how to improve the model further.

Chapter 5 - Summary and Conclusions: Finally, we summarize work done in this thesis and suggests future work to improve the model.

Chapter 2

Background

2.1 Introduction

This chapter provides a concise review of the background knowledge necessary to comprehend the scope of the thesis. We will begin by explaining fundamental concepts in computer vision; more precisely, we will define what a digital image is and how processing operations such as convolution can be utilized to extract useful information from an image. Additionally, we provide a brief explanation of machine learning and demonstrate how we can process images using machine learning techniques such as convolutional neural networks. Additionally, we will discuss additional machine learning techniques such as principal component analysis (PCA) and how it may be utilized to develop a linear model for general face representation known as a 3D morphable model (3DMM). Finally, we will present the metrics used to evaluate our work throughout the thesis.

2.2 Virtual Avatars for Investigative Interviews with Children

As stated in section 1.1, recent research indicates that employing an avatar in interview training may result in improved interview training. The study

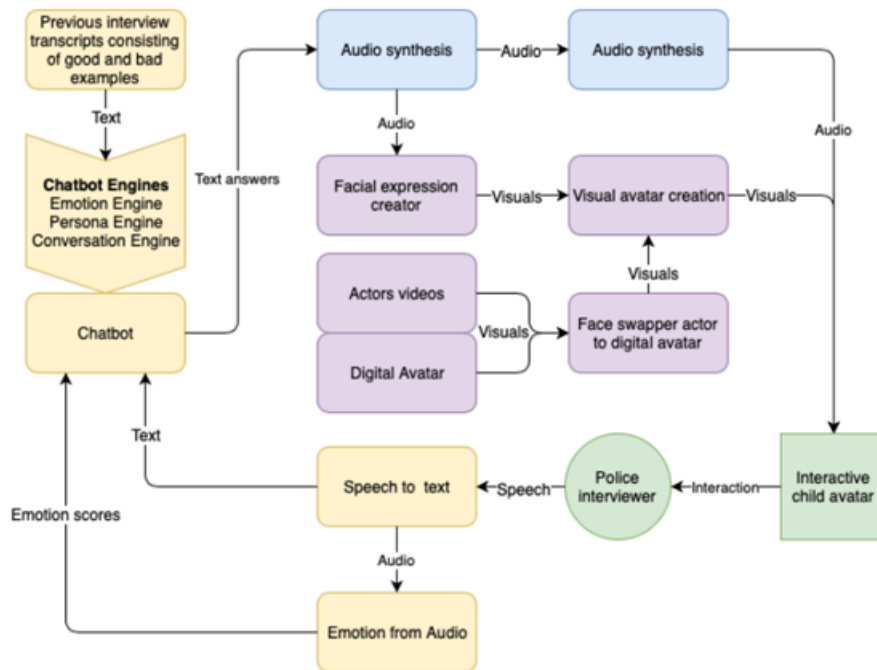


Figure 2.1: Green blocks mark the interactive parts, yellow is text related, blue is audio, and purple marks the part of the system related to the visualization [5].

employed two distinct computer-based child avatars with defined memories regarding child sexual abuse. Combining virtual interviews with avatars and providing regular feedback to participants enhanced interview quality. The participants employed a greater number of recommended questions and a lesser number of possibly harmful strategies [25] [30].

However, prior avatars provided only a limited set of predetermined question sheets the investigator could ask from, which limits the scenarios the training could cover. The multimodal model suggested in [5] (see fig 2.1) describes a system fully capable of automating the whole process of an interactive training regime used to train investigators in conducting interviews with abused children. Using machine learning approaches, the model provides the necessary methods for allowing the trainees to freely interview the avatar as if practicing with a paid actor.

The multimodal model consists of; chatbot, text-to-speech, speech-to-text,

avatar, image manipulator/generator. The chatbot provides the brain to the avatar allowing interactive communication between the trainees and the avatar. The chatbot communicates using text, so the system provides a model for transforming the chatbot output from text to speech and speech to text for the trainees response. By including a speech-to-text and text-to-speech model, the avatar is able to interact with the trainees via audio, making the learning process more practical and realistic. At last, the system provides a model for augmenting the avatar's visual context by giving it a photo-realistic child appearance, with facial movements based on the chatbot's responses during the interview training, while also taking emotion into account, resulting in a real-time re-enactment of a child's facial expressions.

We will concentrate on creating a visual context for the avatar in this thesis; rather than using audio to create facial expressions, we will employ an actor to transfer the actor's facial expression to the digital avatar. Thus, rather of using audio to drive the avatars facial expression, we will use the expressions of an actor.

2.3 Computer vision

Computer vision is a scientific field that focuses on replicating parts of the complexity of the human visual system and enabling computers to identify and process objects in images and videos in the same way that humans do. Due to recent advancements in artificial intelligence, the field of computer vision has been able to surpass humans in some tasks related to detection and labeling.

2.3.1 Digital image

A image may be defined as a two-dimensional function $f(x,y)$, where the x and y are coordinates in the image plane, and the amplitude of f is called the intensity. A digital image is a finite discrete representation of the two-dimensional function f , where each location is occupied by a pixel with its assigned intensity value. Typically, a digital image is represented as a two-dimensional array formed of the numerical value of $f(x,y)$. Computers uses

this format to perform computations, enabling the computer to perform a variety of mathematical operations on the digital image.

The spatial information encapsulated in the array is used in a broad spectrum of image processing applications. For example, a common approach in image processing is to use convolution to extract/filter specific characteristics of a given digital image. Convolution is a mathematical operation on two functions (f and g) that produces a third function ($f * g$) that expresses how the shape of one is modified by the other [41]. In image processing, convolution is performed between the image array and a kernel. The kernel, also referred to as a filter, is a small matrix whose coefficient determines the nature of the resulting characteristic image. Convolution in image processing is accomplished by sliding the kernel across the image, generally starting at the top left corner, in order to traverse all positions where the kernel completely fits within the image's boundaries. Each traverse computes the sum of the element-wise multiplication and assigns it to a corresponding location on the resulting image, see fig 2.2.

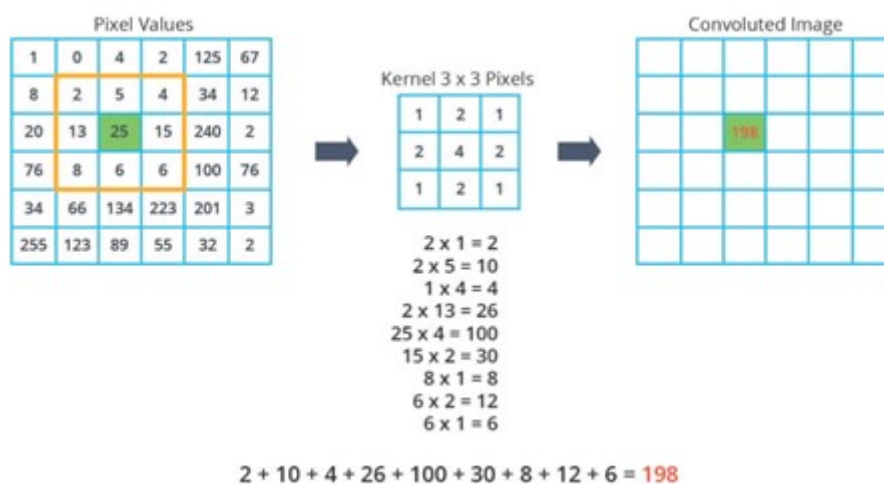


Figure 2.2: Value of a output pixel (right matrix) given as the sum of elements wise product of image window(left matrix, matrix inside yellow boundry) and the kernel(middle matrix).²

²Image taken from: <https://dev.to/sandeepbalachandran/machine-learning-convolution-with-color-images-2p41>

When combined with the suitable kernel, convolution can be used to extract useful properties of an image, such as the image gradient and feature map, which can be utilized for face detection and face parameterization.

2.4 Machine Learning

Machine learning is a branch of artificial intelligence, which leverages statistical modeling and algorithms to solve certain tasks. Machine learning aims to program computers to use example data of past experience to solve a given problem [4], which lets the computer solve tasks based on patterns and inference instead of being explicitly programmed to do so. The fast improvement of computational power and increasing example data in the last decades have contributed to the wide adoption of machine learning in various fields.

Machine learning today is successfully applied to a wide variety of applications used in everyday life, such as translation, search optimization, and image classification. Various approaches use past experience to teach the computer to solve specific tasks. Those approaches are generally divided into three broad categories; supervised learning, unsupervised learning, and reinforcement learning.

2.4.1 Supervised Learning

Supervised learning involves using a set of data containing both input and corresponding labels as our past experience (example data) when learning the computer to solve certain tasks. The goal is to learn a function that maps the input data to the corresponding label. Supervised learning is mainly used in classification and regression tasks, such as image classification and object localization.

Artificial Neural Network

An artificial neural network (ANN) is a machine learning model designed to have our brain's self-learning capabilities by loosely simulating how our brain works with processing data. The architecture of an ANN is loosely based on how our brain has billions of cells called neurons, which make up the processing units of the brain. The neurons are connected by synapses that can transmit a signal to other neurons when activated. Similarly, as the brain processing unit, an ANN is made up of numerous nodes (artificial neurons) that are connected by weights (which simulates the synapses). The nodes are typically aggregated into layers as in fig 2.3. An ANN consists of an input layer, one or more hidden layers, and at last an output layer. Each layer consists of one or more nodes, and the layers are connected by weights, which are the lines connecting the nodes. The weights are trainable parameters, influencing the magnitude of signal transference between a node and nodes in the next layer.

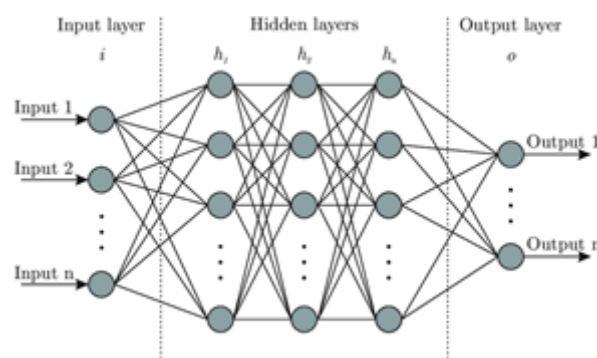


Figure 2.3: A feed-forward neural network.³

The architecture in the fig 2.3 is referred to as a feed-forward network with a fully connected layer, meaning a node is connected to every node in the next layer. The input is a feed-forward network is fed to the next layer where the activation is computed by an activation function (2.1). The activation function decides whether the node should be activated or not by calculating the weighted sum of the weights and input value. Finally, the last layer's activation

³Image from: <https://www.uio.no/studier/emner/matnat/fys/HON1000/v20/studentblogg/veiledet-lering.html>

function is chosen according to the task the network is supposed to solve.

$$a = \sigma \left(\sum_{k=1}^k x_k w_k \right) \quad (2.1)$$

A SoftMax function (2.2) is the most common activation function used in the last layer of an ANN for multi-classification tasks, such as image classification. SoftMax outputs produce a vector that is non-negative and sums to 1, representing the probability distribution for the classes. For regression tasks, a linear activation function is used, as the desired output is an unbounded numerical value.

$$\sigma(x)_j = \frac{e^{z_j}}{\sum_{k=1}^K e^{z_k}} \quad (2.2)$$

The output of an ANN is used with its corresponding label to calculate the loss, which is a numeric value representing the difference between the output and its corresponding label.

The network learns by using backpropagation. Backpropagation is an algorithm used to compute the gradients of the loss with respect to the weights. The gradient describes how much the output of a function changes if the input changes by a little. As a result of using backpropagation, all network operations must be differentiable so that the gradient of the loss with respect to each weight can be computed using the chain rule.

The network weights are updated by using an optimizer. An optimizer is an algorithm used to determine the optimal way to update the weights. A simple approach is gradient descent (GD) [31]. GD uses the computed gradient and the learning rate to determine a step size and direction toward the minima.

Convolutional Neural Network

Convolution neural network (CNN) is a modern neural network architecture popularized in 2012 by winning the ImageNet [12] competition by improving upon the previous images classification error from 26% to 15% of the ImageNet data set. CNN solved a lot of the issues associated with feed-forward neural networks. CNN showed significant improvements in preserving spatial

features and generalizing. In a feed-forward neural network, the input data passed into the neural network had to be transformed into a one-dimensional vector to fit into the network, resulting in critical spatial relationships in multi-dimension data diminishing. Take a 256X256x3 shaped image as an example; when transformed into a 1d vector would result in a vector containing 196 608 elements. Given that every node in the current layer is connected to every node in the next layer, the amount of parameters would quickly become computationally unmanageable.

CNN introduces an excellent solution to those issues and is today incorporated in a wide variety of state-of-the-art neural network models. Unlike feed-forward neural network, an CNN takes in a multidimensional vector in the form of an image as an input, which allows for encoding specific properties into the architecture. As in a feed-forward network, CNN consists of an input layer, a hidden layer, and an output layer but is arranged differently, see fig 2.4. Unlike feed-forward neural networks, which only consists of fully connected layers, CNN usually incorporate two new types of layers to the model architecture, resulting in an architecture consisting of convolutional layers, pooling layers, and a fully connected layer.

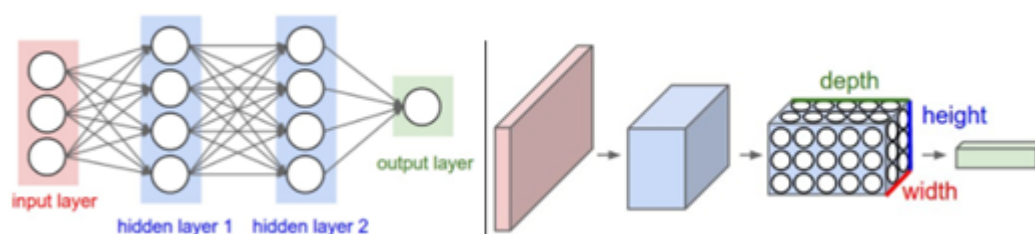


Figure 2.4: A CNN arranges its neurons in three dimensions (width, height, depth), as visualized in one of the layers. Every layer of a CNN transforms the 3D input volume to a 3D output volume of neuron activations. In this example, the red input layer holds the image, so its width and height would be the dimensions of the image, and the depth would be 3 (Red, Green, Blue channels)[1].

As the name suggests, a convolutional layer uses convolution in place of general matrix multiplication. As described in section 2.3.1, a filter W (kernel)

is slid across the height and width of the layer computing the dot product between the filter and the layer at every position, see fig 2.2. In this case, the filter is learnable and extends through the whole depth of the source layer. The convolution between the filter and convolutional layer is followed by an activation function resulting in a 2d activation map that will be the input on the next layer. The main goal of the convolution operation is to extract good features from the input image. Convolutional neural networks usually have several convolutional layers. Conventionally, the first convolutional layer takes on the task of identifying and retaining low-level information such as edges, color, corners, etc. With the addition of new convolutional layers, the architecture shifts in response to learning high-level characteristics, giving us a network, which exhibits a complete understanding of the image.

The big selling point of the convolution layers is the sparse interaction between the layers; the output from the layer is only connected by a few neighboring activations from the previous layer, which are referred to as the receptive field. The receptive field is defined as the region in the input that a particular output node is affected by. This ability reduces the parameter size significantly, allowing the network to generalize much better. The neighboring manner of the connection also improves learning the characteristic of the input image. The neighboring pixels are usually more correlated; therefore, each activation in the output layer represents a small area which encourages the network to learn more spatial features.

Pooling layers lower the size of the incoming input (activation from previous layers) to reduce the computational workload. The usual size is 2×2 , which will reduce four activation inputs into one output. By applying spatial pooling, the network will still retain important spatial features. Some common types of spatial pooling are max-pooling, average-pooling, and sum-pooling. As the max-pooling suggests, max-pooling reduced a 2×2 activation area by retaining the highest value see fig 2.5. In contrast, average pooling reduces by taking the average, and sum-pooling does by reduction by summing.

The fully connected layer is usually the last layer in a CNN, which operates as a classifier mapping the extracted features from the previous layer into a class or a class distribution. The network learns the same way as feed-forward

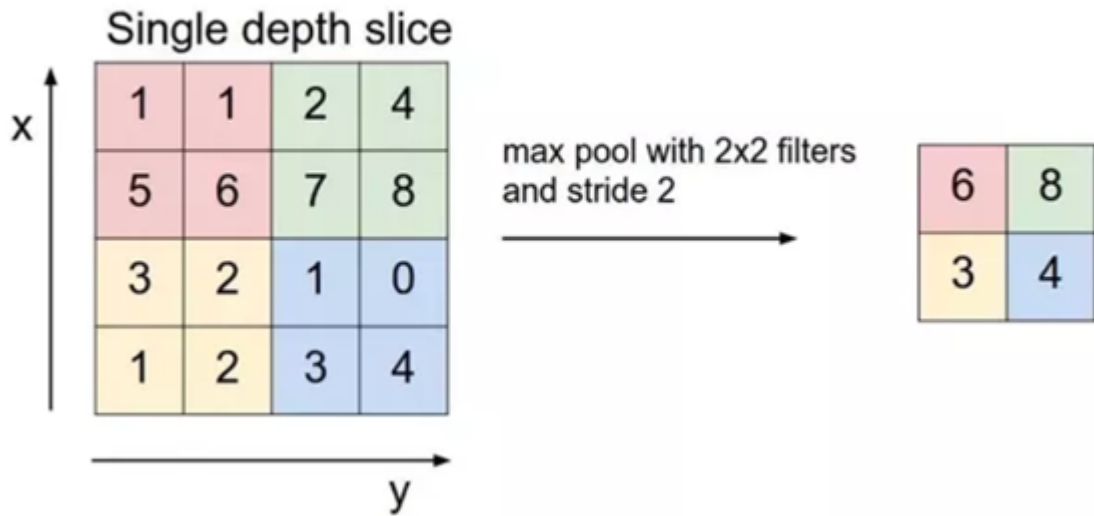


Figure 2.5: Max pooling performed on the matrix to the left, the matrix is subdivided using a 2x2 filter and stride 2 and the max value is selected at each subsection. Stride is the number of pixels by which the window moves after each operation.

network, by backpropagating the loss and updating the filter parameter (weights).

2.4.2 Unsupervised learning

Unsupervised learning enables us to work with a data set without requiring associated labels, which is time costly or hard to collect in a various situations. Unlike supervised learning, Unsupervised learning involves using a data set without corresponding labels. Unsupervised learning is traditionally used to find or leverage the underlying patterns in the input data, such as in Generative adversarial network [18] and Principal Component Analysis [24].

Principal Component Analysis (PCA) is a dimensionality-reduction method that is often used to reduce the dimensionality of large data sets by transforming a large set of variables into a smaller one that still contains most of the information in the large set.

PCA is defined as an orthogonal linear transformation that transforms the data to a new coordinate system such that the greatest variance by some scalar projection of the data comes to lie on the first coordinate (called the first principal component), the second greatest variance on the second coordinate, and so on.

2.4.3 3D Morphable Face Models

3D morphable model can be thought of as a 3D extension of morphing; morphing is a special effect used in film and animation that seamlessly transforms one image or shape into another. Thus, the term 3D morphable hints to the capability of seamlessly transitioning from one 3D shape to another.

A 3D Morphable Face Model (3DMM) [15] is a generative model for determining the shape and appearance of a face based on two fundamental concepts: To begin, all faces are in dense point-to-point correspondence, see fig 2.6, which is established during training, where a large number of faces are registered and further preprocessed such that each entry point in the data vector containing the dense face points corresponds to the same point on the other faces.

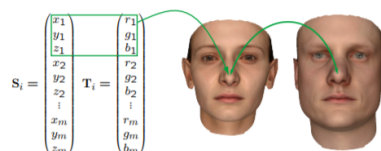


Figure 2.6: Each point Each entry in the data vectors correspond to the same point on the faces. In this example the first entry corresponds to the tip of the nose [29]

Second, the collected faces are used in a statistical model such that a new face can be generated through some form of linear combination between the collected faces. This is possible due to the point correspondences between the collected faces.

Nota bene, the 3DMM also includes an appearance (texture) model, but this thesis will focus on the shape model.

Basel face model

The Basel face model [29] (BFM) is a 3DMM based on PCA and trained on a training set with large variety of shapes. The BFM training data set comprises face scans of 100 female and 100 male subjects with age and weight distributed over large rang, see fig 2.7. Each subject was 3D scanned three times with a neutral expression, and the scan with the most natural look was added to the training set.

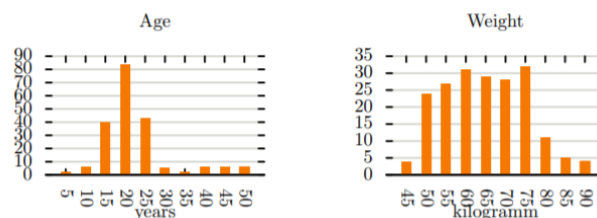


Figure 2.7: The BFM was trained on 200 individuals (100f/ 100m). Age (avg 25y) and weight (avg 66kg) are distributed over a large range, but peak at students age[29].

The shape of the collected faces are represented by the 3m dimensional vector

$$S = (x_1, y_2, z_3, \dots, x_m, y_m, z_m) \quad (2.3)$$

where m is the number of vertices. The vertices are used as the corners in triangles which are grouped to create a mesh. The generated face meshes share a common triangulation such that the triangles connections only need to be generated once.

The BFM model generates a new novel face by a linear combination of the average face and the product of the d most dominant eigenvector of the covariance matrix computed over the shape difference $S_i - \bar{S}_i$ and a low-dimensional identity parameter; note that the d most dominant eigenvectors of

the covariance matrix are the principal components from PCA. Using the PCA, the training data is fit to the model given as

$$S(\alpha) = \bar{S}_{id} + \mathbf{E}_{id}\alpha \quad (2.4)$$

where \bar{S}_{id} is the computed mean shape over the training set, \mathbf{E}_{id} is the principal components, and α is the low-dimensional identity parameter.

The BFM model can be extended to include a model for expression using a training set from FaceWarehouse [8]. The FaceWarehouse data set consists of face scans of different faces where each face is scanned in 20 different expressions, including one in a neutral state. Then a new data set is composed by the offset between expressive scans and neutral scans: $\Delta S_{i,e} = S_{i,e} - S_{i,0}$ where i yields the number of different people scanned while e yields the number of different expressions scanned, and $e=0$ is neutral face scan. Using the PCA, the new model can be expressed as

$$S(\beta) = \bar{S}_{exp} + \mathbf{E}_{exp}\beta \quad (2.5)$$

Combining 2.4 and 2.5, a new model can be used to generate a new face with identity shape and expression variations, the combined model is given as

$$S(\alpha, \beta) = \bar{S} + \mathbf{E}_{id}\alpha + \mathbf{E}_{exp}\beta \quad (2.6)$$

where $\bar{S} = \bar{S}_{id} + \bar{S}_{exp}$.

2.5 Metrics

In this section we describe the metrics used to evaluate our final model in this thesis. We evaluate the models' ability to synthesize photo realistic avatars with its given input expression using two perceptual similarity metrics. We provide a metric for estimating inaccuracy for use in the parameterization of the input portrait using pre-trained models. Additionally, we provide two metrics introduced in [11] for evaluating the models ability to preserve temporal quality.

2.5.1 L2 distance

The $L2$ distance [27] (often referred to as Euclidean distance) is the shortest distance between two points in a euclidean space. It is a widely used metric for determining the similarity of two data points and is utilized in various domains, including mathematics, physics, and machine learning. $L2$ distance is calculated from the Cartesian coordinates of the points using the Pythagorean theorem and is given as

$$L2(p, q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \cdots + (p_i - q_i)^2 + \cdots + (p_n - q_n)^2} \quad (2.7)$$

for the two point p and q , where n is the number of dimensions [37].

2.5.2 Normalized Mean Error

Normalized mean error (NME) is used to assess the quality of the estimator used for extracting the face features. The NME used in this thesis is a combination of the mean error described in [17] and the normalization method used in [7]. Normalization allows for comparing results from different faces on a standardized scale. NME is given as

$$\text{NME} = \frac{1}{N} \sum_{k=1}^N \frac{\|\mathbf{x}_k - \mathbf{y}_k\|_2}{d} \quad (2.8)$$

where x is the estimated value, y is true value and d is the normalization factor given by square-root of the ground truth bounding box (the bounding box used to crop the face to fit the 3DDFA model) $\sqrt{w_{bbox} * h_{bbox}}$.

2.5.3 Learned Perceptual Image Patch Similarity

Learned Perceptual image patch similarity (LPIPS) [42] is used to compare the similarity between two different images, **LPIPS** works well on images that look similar, like the same images but shifted by one pixel or blurred. LPIPS calculates the similarity by first computing the distance between the two different activations at each layer caused by running the images through

a pre-trained CNN and then computing the similarity, based on the difference in activations. The pre-trained CNN used is a VGG-trained ImageNet classifier [32] [12], which excels at feature extraction.

2.5.4 Structural Similarity Index Measure

Structural similarity index measure (SSIM) [40] is a method of evaluating perceptual image quality; many image quality metrics are based on comparing the values of pixels between a reference and a sample. Unfortunately, this can frequently be deceptive. For example, the human vision will think of images in which the reference image has been shifted a small number of pixels to appear nearly identical to the reference image. However, a method that measures the difference in the values of each of the corresponding pixels between the shifted reference and reference images will report a significant difference when in reality (as envisioned by the human vision) is relatively small. SSIM is based on simulating human vision, which can extract structural information from a scene and thus distinguish between the information extracted from a reference and a sample image.

The SSIM score is often adjusted between 0 and 1, with a value closer to 0 indicating that the reference and sample image are quite dissimilar and a value closer to 1 indicating that the images are very similar.

2.6 Temporal Optical Flow

Temporal optical Flow (tOF) measures the pixel-wise difference of motions estimated from sequences. We differentiate the optical flow estimation of the target and output video to compute the tOF. Optical flow is defined as the distribution of apparent velocities of movement of brightness pattern in an image [21]. The Farneback's algorithm [16] is used to estimate the optical flow. The tOF is given as follow

$$tOF = ||OF(x_{t-1}, x_t) - OF(y_{t-1}, y_t)||_2 \quad (2.9)$$

Where x_t is target at time t , and y_t is output at time t and OF is the optical flow.

2.7 Temporal Learned Perceptual Image Patch Similarity

Temporal learned perceptual image patch similarity (tLP) measures perceptual changes over time using deep feature map, we compute the tLP by measuring the difference between perceptual change in target video and output video. The tLP is given as follow

$$tLP = ||LPIPS(x_{t-1}, x_t) - LPIPS(y_{t-1}, y_t)||_2 \quad (2.10)$$

Where x_t is target at time t , and y_t is output at time t and LPIPS is the LPIPS is the Learned Perceptual image patch similarity.

2.8 Summary

We utilized this chapter to provide all the background information essential to grasp the methodology chapter that follows. The required background knowledge can be divided into three major sections. First, we discussed how we can use machine learning, more precisely CNN, to extract information from an image by leveraging prior knowledge from the trained convolutional neural network, such that the trained weights in the CNN can be used to create an activation map for the useful feature in the input image, and how those features can be used in classification or regression tasks.

Second, we described how to apply PCA to generate a linear model for producing a 3D face composed of vertices and the edges that connect them. Additionally, we reviewed the 3DMM used in the next chapter and how it can be parameterized using identity and expression parameters to construct a 3D face with the desired shape (a combination of identity shape and expression). Finally, we described some metrics for evaluating various aspects of an estimator's performance or comparing different data.

In next chapter will address how we can utilize a CNN to transform an image portrait into the 3DMM parameters identity and expression, and

then combine those parameters with another set of identity and expression parameters to generate a new face. In practice, we combine portraits by encoding them into a lower-dimensional representation and then decoding them back into a higher-dimensional representation in the form of a new portrait.

Chapter 3

Methodology

This chapter describes in detail how the facial avatar algorithm is implemented. The overall method incorporates a 3D morphable model, synthesizers, a renderer, and a blender to create an end-to-end solution for reenacting a facial expression from a source avatar to a target avatar, see fig 3.1. The method is restricted to only reenacting the lower facial expression, maintaining the original target face pose and identity.

This implementation's overall purpose is to create an avatar video of a talking face utilizing features extracted from a source video of a talking face. The extracted feature from the source video is utilized to build an avatar with the same facial features as features extracted from the source video, such that the synthesized avatar has the same facial expression (mainly around the mouth region) as the face captured in the source video. In addition, the avatar's face identity shape and texture will be derived from a target video of a talking face. Thus, the avatar is essentially a composite of traits derived from the source frame, target frame, and a collection of frames from the target video, with the expression belonging to the source frame, the identity shape belonging to the target frame, and the lower face texture belonging to a collection of frames from the target video.

The end-to-end algorithm can be subclassed to 4 major methods consisting of:

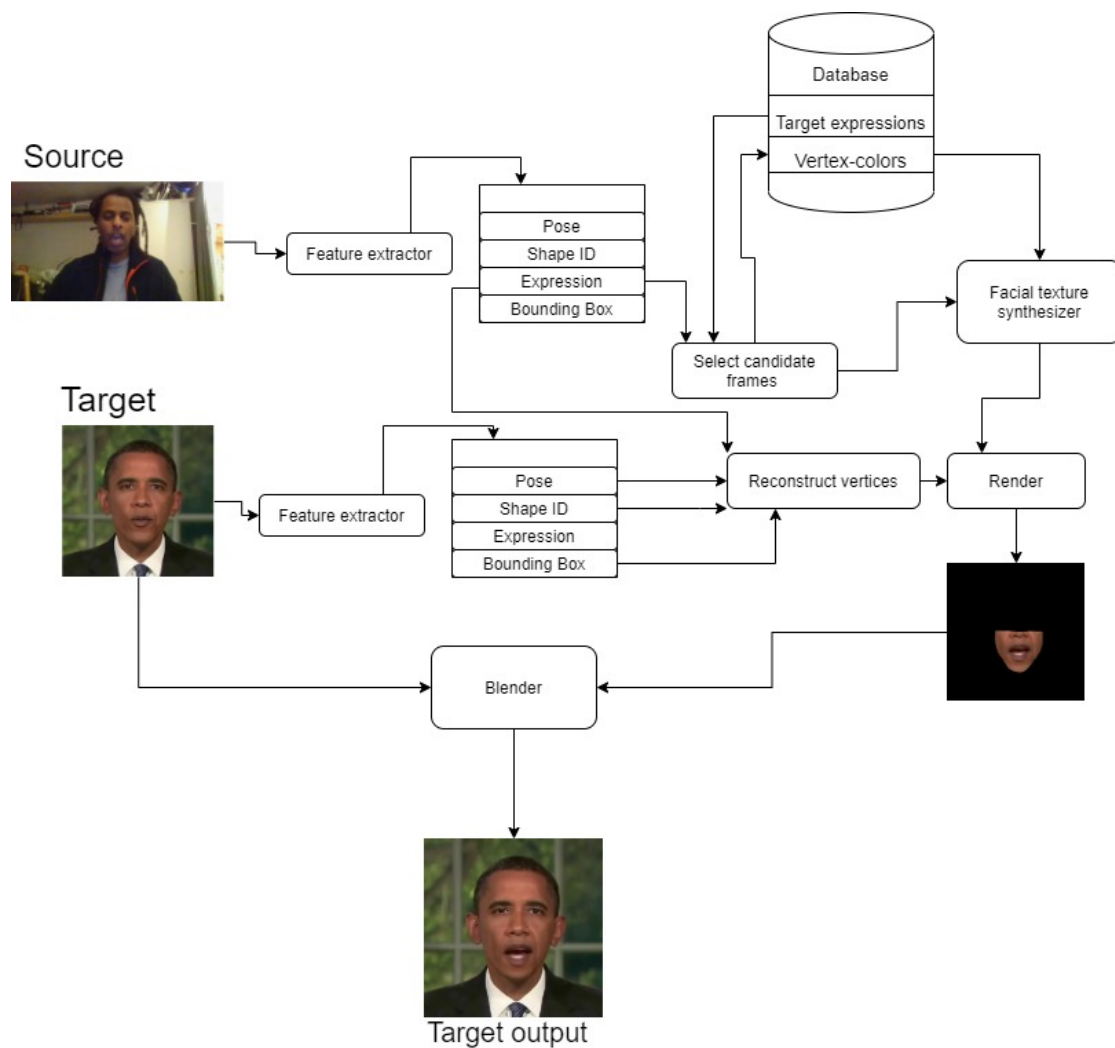


Figure 3.1: Overview of the model from input (source,target) to target output

Feature extraction - Extracts 3DMM parameter from the source frame, which are used to create mesh of the source face, later reduced to a mesh of the mouth region.

Facial texture synthesizer - Uses the parameter from the feature extractor to predict a suitable texture for the meshed face generated in the previous method.

Renderer - Renders the lower mouth region using the predicted colors and

constructed mesh

Blender - Blends the rendered mouth region with the target video

3.1 Feature extractor

The feature extractor used to parameterize the source face is a 3D dense alignment method from [19] referred to as 3DDFA. 3DDFA is a machine learning based regression network that achieves state-of-the-art performance at the time of development. The model uses a lightweight network architecture that allows the model to run at 50fps on a single-core CPU. The lightweight backbone used in the model is the a MobileNet [22] architecture, which is based on depth separable convolution [10] resulting in a CNN model with less computation and parameters.

The 3DDFA model accepts an image as input and regresses to the 3DMM parameters needed to reconstruct the 3D face mesh. Prior to the image being passed through, the face region is cropped using a face detector. The face detector module utilized is called faceboxes [43], and it is based on a CNN model and performs well in terms of accuracy and efficiency.

The cropped input image is regressed to the target parameter \mathbf{P} which is given as $\mathbf{P} = [\mathbf{K}, \alpha, \beta]$ where \mathbf{K} is the camera matrix (shape of 3x4), α is the identity parameters (shape of 40x1), and β is the expression parameter (β , shape of 10x1), with 62 dimensions in total. α and β are used with the linear model eq. (2.6) to project the input 2D face to a dense 3D face in canonical space, and the camera matrix is then used to match the pose of the projected 3d face with the 2d input face. Overall the full projection from the 2D face to the 3D face is given as

$$S(\mathbf{P}) = \mathbf{K} \begin{bmatrix} \bar{\mathbf{S}} + \mathbf{E}_{shape}\alpha + \mathbf{E}_{exp}\beta \\ 1 \end{bmatrix} \quad (3.1)$$

The projected dense 3D face can easily be projected back to the image space using orthogonal projection, the projection back from the image plane is given as

$$S_{2d}(\mathbf{P}) = \mathbf{Pr} * S(\mathbf{P}) \quad (3.2)$$

where \mathbf{Pr} is the orthogonal projection matrix $\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$.

The projected 3D face consists of vertices used to generate the 3D face mesh using a predefined triangulation. The triangulation is shared between all the face meshes. Once the 3D face mesh is constructed, the mesh is reduced to only cover the lower mouth/face region. The region is decided using a handmade mask made with MS-paint¹, the vertices inside the mask and its associated faces are kept while discarding the remainders.

The 3d Basel mode used for reconstruction does not cover the mouth region due to ambiguous geometry in that region. Therefore, the empty space inside the mouth must be filled manually, see fig 3.2. This process will have to be done in a deterministic fashion to ensure that vertices inside the empty region have somewhat good correlations between different images. The empty mouth space is filled accordingly:

1. First, we determine the bound vertices of the empty mouth space, this is only needed to be done once. Once we have the bound vertices, we can save the indexes to be later used on new set of vertices.
2. Use the bound vertices to calculate the mean point
3. Draw N evenly spaced points/vertices from all the mouth bound vertices to the mean point
4. Use Delaunay triangulation² to generate mesh for the vertices drawn.

¹Microsoft paint: https://en.wikipedia.org/wiki/Microsoft_Paint

²Scipy: <https://docs.scipy.org/doc/scipy/reference/generated/scipy.spatial.Delaunay.html>

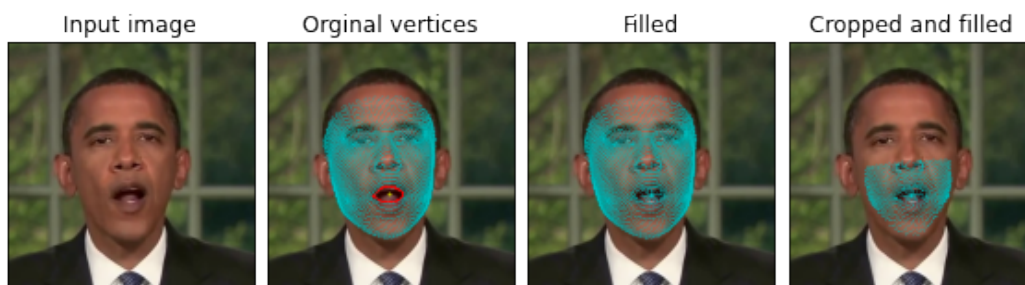


Figure 3.2: Extending the BFM model to fill mouth and cropping to fit lower face. The red contour in original vertices shows the boundary of the non-filled mouth area, while the yellow dot in that image shows the midpoint of the unfilled area.

3.2 Facial texture synthesizer

The facial texture synthesizer is based on leveraging existing images to synthesize a new image. For this part, the data set consists of a large number of frames (5000+) of the desired target person. The collected data is from a monocular sequence video of the target person talking and lightly moving his/her face; for this example, a video consisting of 8000 frames of Obama will be used. The video can be found on the following GitHub repository³. Furthermore, the quality is relatively high (450x450x3,RGB color), with the face region occupying a relatively large part of the frame and the lighting and composition staying coherent. The video is upscaled to 512x512 because of the blending method that will be used further in the process. The upscaled video goes through a preprocessing, where the 3D face parameters and the vertex colors are collected, see fig 3.3.

³Video: <https://github.com/YudongGuo/AD-NeRF/tree/master/dataset/vids>

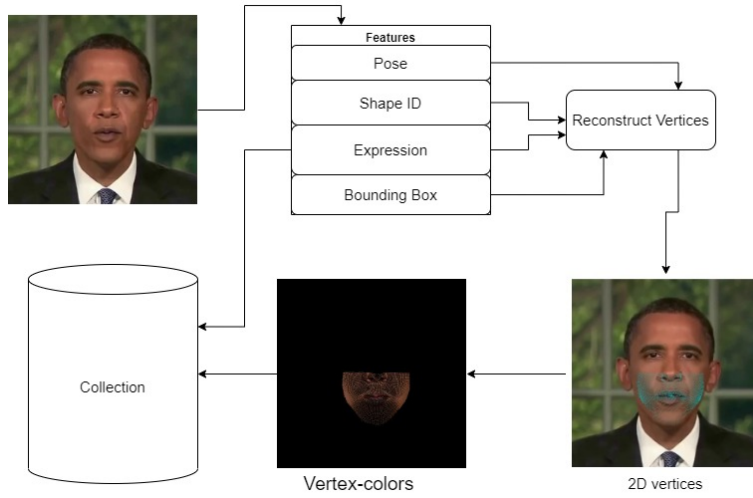


Figure 3.3: The preprocessing method used to collect samples data

For each frame in the target video, the follow preprocessing is done:

1. The face parameters, and lower face vertices are collected using the methods described in the previews section
2. The collected vertices are projected back to the image plane using the eq. 3.2, and are then used to sample the color at the vertex position in the image being preprocessed, this will be referred to as the vertex-color.

Once the target video(data set) is preprocessed, the collected data are stored in an array. The resulting arrays are an $[8000,62]$ (n-frames, \mathbf{P}) array storing the face parameters, an $[8000,4]$ array storing the corresponding bounding box of the face detected, and at last an $[8000,10011,3]$ (n-frames, vertex-color, channels) array. Finally, the stored arrays are used to synthesize the desired face.

Given the extracted face parameters from the source frame (input image), the L_2 distance is calculated between the source expression parameter and all the expression parameters stored in the data set. A fixed number of frames from the data set that has the smallest L_2 distance are selected as the candidate frames. Once the n best candidate frames are chosen, the associated vertex-colors are used to compute the weighted median. The weights are chosen accordingly:

$$w_i = e^{-\frac{|x_i - x_s|_2^2}{2\sigma^2}} \quad (3.3)$$

where w_i is the weight at i th frame, x_i is the expression parameters from stored data set and x_s is the source expression parameter. The σ in the weights contributes to weight tuning. Small σ results in a peak distribution, indicating a high contribution from a few frames, resulting in temporal flickering, whereas large σ results in a smoothed distribution, indicating a blurry result due to the high contribution from many frames.

The optimal sigma for the given expression parameter can be suboptimal for another expression parameter depending on the number of good candidates, i.e., ones with a small $L2$ distance. σ is selected adaptively, since the optimal σ is proportional to the number of good candidates. The optimal σ can be selected by finding σ such that the weight contribution of n candidates is a -fraction of the weight of all available frames. In other words, the optimal σ can be found by solving where the sum of n candidate weight is close to equal to the sum of a -fraction weight of all available frames N in the stored data set, the function to be solved is given as:

$$\sum_{i=1}^n w_i(\sigma) = a \sum_{i=1}^N w_i(\sigma) \quad (3.4)$$

The equation (3.4) is solved using a binary search on σ , where $a=0.9$.

The weighted median is used to compute the texture using the vertex color associated with the chosen candidate frames. Using the weighted median gives the advantage of realistic texture synthesizing while avoiding too much temporal flickering; this is due to its ability to preserve central tendency while avoiding too much contribution from outlier frames. The weighted median is computed by sorting the pixels stored at each vertex according to their weights and choosing the pixel situated at half of the total weight, see listing 1.

```

@numba.njit(parallel=True)
def wm(colrs,weights):
    n_frames,n_verts,channels=colrs.shape
    out=np.empty((n_verts,channels),dtype=np.float32)
    for i in prange(n_verts):
        for j in prange(channels):
            data=colrs[:,i,j] # get pixels in n_frames at [i,j]
            ind_sorted = np.argsort(data) #argsort by pixel value
            sorted_data = data[ind_sorted] # sort data using pixel value
            sorted_weights = weights[ind_sorted] # sort weights
            Sn = np.cumsum(sorted_weights) # cumulative sum

            # find value middle point of the cumulative sum
            Pn = (Sn-0.5*sorted_weights)/Sn[-1]
            I=np.searchsorted(Pn, 0.5) # find index value
            # interpolate to find middle value pixel value
            out[i,j]=((sorted_data[I]-sorted_data[I-1])/
                    (Pn[I]-Pn[I-1]))*(0.5-Pn[I-1])+sorted_data[I-1]

    return out

```

Listing 1: Weighted median calculation in python

The target vertices are reconstructed using eq.2.6 where β is the expression parameter from the source frame, and α is the identity parameter from the target frame. The reconstructed vertices are then cropped and filled to fit the lower mouth region, then used to generate the lower face mesh. Finally, the generated mesh with the calculated vertex colors renders the synthesized lower face region. Note that this method closely resembles the method used in [35] to synthesize Obama, but there are two distinct differences:

1. The method in [35] uses the mouth key-point landmarks to select the candidate frames, while we use the expression parameter.
2. The preprocessed data set consists of frontalized images of the target person which are used to calculate the weighted median. This is possible due to that fronalization contributes to the make pixel entries loosely correspond between the faces. We are using the 3DMM for the purpose of aligning the pixels entries between the faces.

3.3 Renderer

A rasterization algorithm is used to render the synthesized image from the constructed lower face mesh and its associated vertex colors. Rasterization is the most common rendering technique used to render images of 3D scenes. The framework/library used in this project is a lightweight implementation of rasterization with z-buffering optimization written mainly in c++ and exported to Python with Cython extension ⁴. The implementation is fast, and it runs on a CPU, avoiding further GPU dependence, making it ideal for this situation.

This algorithm can roughly be decomposed into three steps; projecting the given 3D vertices making up triangles onto the image plane using camera projection, looping through the output frame detecting whether the pixel lies within the resulting 2D triangles projected to the image plane, and finally coloring the visible pixels using the vertex colors to estimate the pixel value and using z-buffering to check if the pixels is visible.

The 3DDFA model uses an orthogonal projection when reconstructing the vertices, making the rendering quite simple. Even though orthogonal projection does not have the perspective preserving perks of perspective projection, it still provides a good result for the critical region cause of low variation in depth relative to the rest of the scene.

3.4 Blender

Once the lower face is rendered, the rendered lower face is combined with the target face to synthesize the new face, which combines the identity from the target face and the expression from the source face.

The composite is done using the Laplacian pyramid blending method. Laplacian pyramid blending [36] allows for smooth blending between the rendered lower face and the target face. Laplacian pyramid blending consists of three steps; pyramid decomposition, pyramid blending, and pyramid reconstruction.

⁴Source: https://github.com/cleardusk/3DDFA_V2/tree/master/Sim3DR

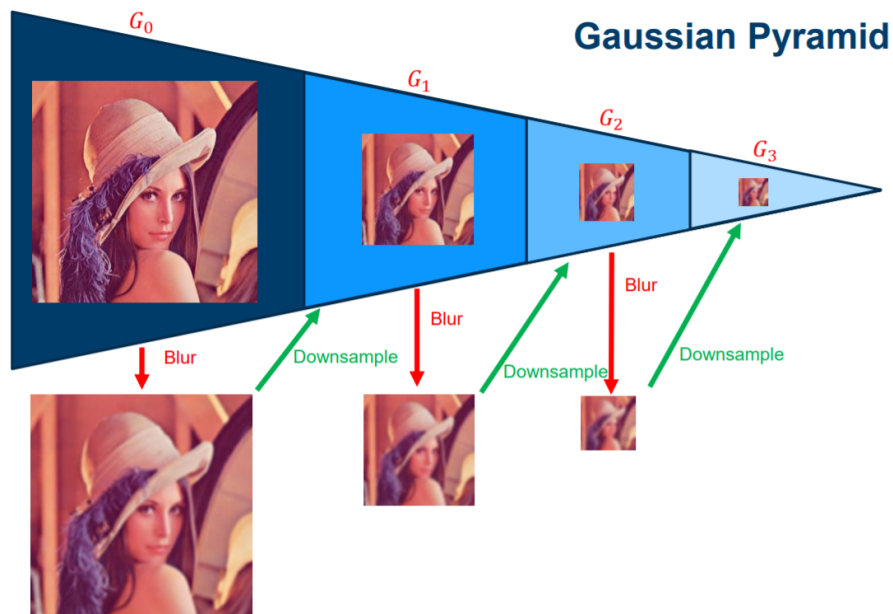


Figure 3.4: Gaussian pyramid [2]

Given two images (synthesized lower mouth and target frame) and a weight mask, the images and mask are decomposed into Gaussian pyramids; the Gaussian pyramids are sequences of images starting from the original and downscaled using subsampling into half size for each step upward in the pyramid, see fig 3.4. The downsampling is done by first convolving the image with a Gaussian filter to avoid antialiasing and then sampling by keeping every 2nd pixel in the convolved image; this is equivalent to performing 2D convolution with a Gaussian kernel and stride size of 2.

The Gaussian pyramid is used to display image information at various scales while retaining the original image information. Gaussian filtering reduces antialiasing by smoothing the image, therefore reducing the high frequency in the image generated by rapid changes in pixel values, which are amplified by subsampling since the subsampling reduces the pixel distance between pixels.

The Gaussian pyramid is used to reconstruct the Laplacian pyramid, see fig 3.5. Given the Gaussian pyramid G , the Laplacian image at pyramid step i is constructed by subtracting the rescaled Gaussian image from its previous

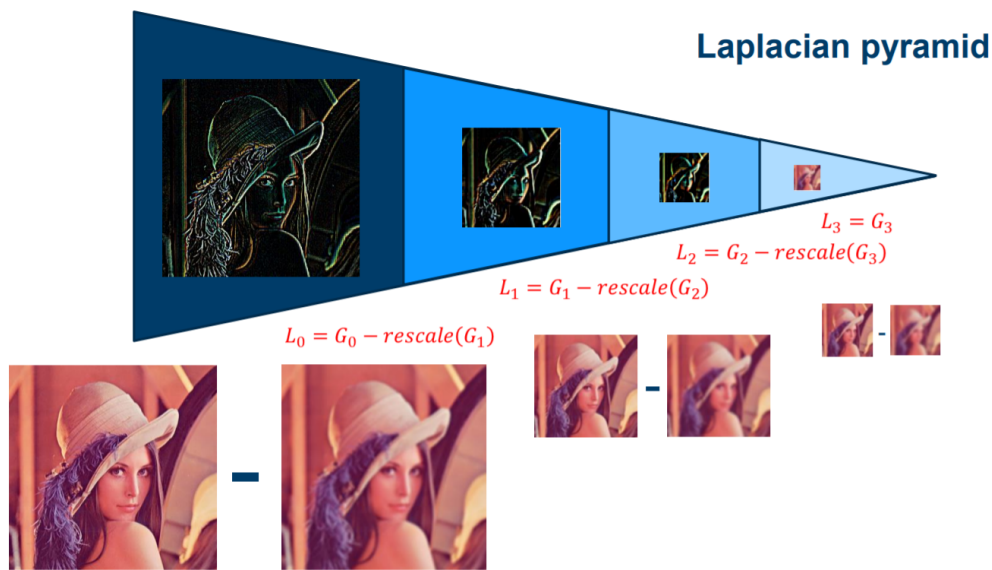


Figure 3.5: Laplacian pyramid [2]

Gaussian image, resulting in the Laplacian image at each step i equaling $G_i - \text{rescaled}(G_{i+1})$. The Laplacian pyramid is constructed for the target image and the rendered image and used to save the approximated high-frequency information lost during sub-sampling.

Once the Laplacian pyramids are reconstructed, the images at each step of the pyramids are multiplied by their corresponding gaussian mask and added together, constructing a pyramid with the composite images at each step. At last, the pyramid is collapsed by upscaling, starting from the top and adding to the lower step of the pyramid until the bottom step is reached. The key to the Laplacian blending approach is that the low-frequency color variations of the rendered image and target image are smoothly blended, while higher-frequency textures are blended more rapidly to minimize "ghosting" effects when the two images are layered, see fig 3.6 for seamless blend using Laplacian Pyramide.

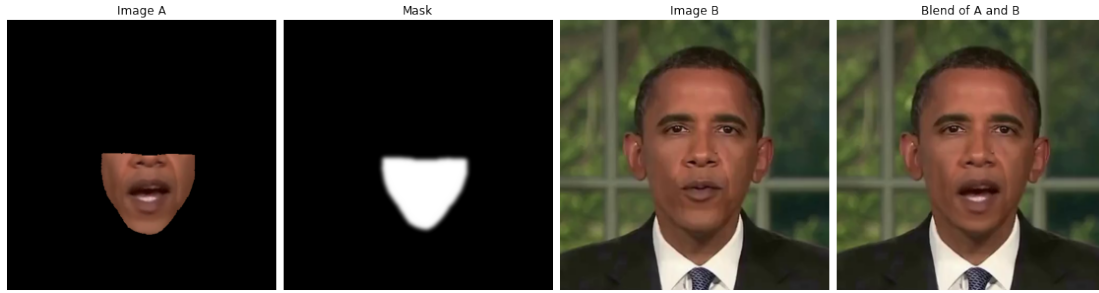


Figure 3.6: Example of a blend of image A and B, Image A is blended to Image B using the mask

3.5 Smoothing

In the real world, a natural mouth motion is generally restricted by smooth gradients, and the motion vector tends to follow a smooth path. Unlike in real life, the parameterization error in our model affects the mouth movements of the avatar speaking. The noise/error in the reconstructed vertices causes the motion of the synthesized mouth to look unnatural, and typically it tends to make the motion vectors rapidly change direction if the noise of the reconstruction error is larger than the average mouth displacement (change in mouth position from frame i to $(i + 1)$). This will result in the mouth motion appearing shivering. To combat this effect, we will use a simple gradient smoothing technique. This is simply implemented by using averaged vertices over time T ; our new vertices are defined as

$$V_i = \frac{1}{T} \sum_{n=i-T/2}^{i+T/2} V_n \quad (3.5)$$

This can also be applied to the vertex colors to smoothen pixel value changes between frames.

3.6 summary

This chapter describes the many methods used to develop a system for reenacting the source expression on the target face. Our system is defined

by four primary methods: a feature extractor, a facial texture synthesizer, a renderer, and a blender. The feature extractor parameterizes the source and target face to the 3DMM parameters identity, expression, and pose using three pre-trained machine learning models. Such that for each source frame, we synthesize a target facial texture by finding the n closest target expression to the source expression. The n closest target expression and their associated vertex colors, which we pre-collect, are used to synthesize the facial texture using a weighted median to combine the n closest vertex colors using their associated expression as weights. After computing the texture, the texture is used to render and blend the texture into the target face, resulting in a target face with the source expression.

Chapter 4

Experiments and results

4.1 Subjective comparison of models

In order to test the facial texture synthesizer models, we have performed a subjective study. We have generated two clips, one for each model, where a person speaks out the alphabet, and the models generate the visual phase of the avatar.

4.1.1 Data

The data used consist of 3 monocular video sequences (see fig 4.1) where a person speaks to the camera recording. Two of the videos (Obama and child) are used as the target data to generate the avatar, which is being reenacted upon, while the last video (source) is used as the source video driving the reenactment of the target avatar. Both the source and child video is self-recorded for this thesis, while the Obama video is downloaded from Github¹. The source video features my utterances of the alphabet, captured using a web camera. The child video has a family member reading aloud from a book and was captured with an iPhone. The Obama video features Obama addressing a political issue; it is identical to the video used in[20]. More info regarding the data set is given in the following table 4.1.

¹Video: <https://github.com/YudongGuo/AD-NeRF/tree/master/dataset/vids>

Data set	Gender	Age	Video quality	fps	Length(frames)
Source	Male	27	Poor	30	500
Obama	Male	65	Good	30	8000
Child	Male	10	Average	30	9100

Table 4.1: Data set used in testing

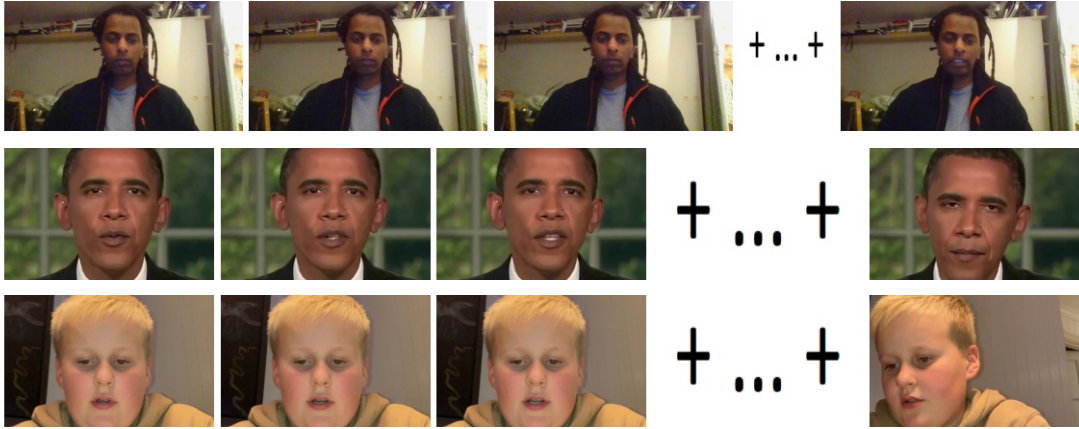


Figure 4.1: Data set used in this case study, $n=1-500$ is used as target background for Obama and child

Both target videos are subdivided into two sections. The first section consists of the first 500 frames and serves as the target frame, which is used together with the synthesized lower face region to composite the final output frame. Additionally, for the first section, the bounding box, the identity parameter α_{target} and pose are extracted using the 3DDFA model and face detector. The second section consists of the remaining frames after subtracting the first section and undergoes the preprocessing method described in 2, which includes collecting the expression parameters β_{target} and vertex colors. For the source video, the expression parameter β_{source} is collected using the 3DDFA model. We also keep the audio from the source video.

The preprocessed data set is used in two similar models with slight variation. The general model is described accordingly;

1. For each frame in the source video select n ($n = 100$) best candidates frames from the target video using β_{source} and all the β_{target} .

2. Compute the lower facial region texture using the method described in section 3.2.
3. Reconstruct the target vertices using equation (3.2) with α_{source} , β_{target} and target pose as input.
4. Crop vertices, then fill the empty mouth region according to section 3.1.
5. Project the reconstructed target vertices to the target image plane using equation (3.2).
6. Render the lower face using the computed lower facial region texture, the projected target vertices and its associated triangulation.
7. Blend the target frame and the synthesized lower facial region.

The first method used is equal to the general method, while the second method implements the smoothing technique described in section 3.5 to the general method. Both methods incorporate the audio from the source video into the created snippets.

The test includes 5 participants, where 3 are females, and the rest are males. The age of participants spans from 20-50, where 2 range from 30-40, other 2 range from 40-50, and the last one range from 20-30.

4.1.2 Questionnaire

The following questionnaire asks participants to rate the generated clips; the questionnaire is created using Google Forms, and the complete questionnaire is included in the appendix. The questions that are posed to the participants can be separated into six groups. The first portion contains questions about the model's overall performance, in which participants are asked to rate the model on a scale of 1-5 (bad, poor, fair, good, exceptional) for Q1 (question number) and scale of (strongly disagree, disagree, neutral, agree, strongly agree) for Q2 and Q3, see table 4.2.

Q	Question from first portion
Q1	How would you rate the overall video quality?
Q2	The visual appearance of the video is realistic.
Q3	The voice appears convincingly to be coming from the person in the video

Table 4.2: Question set 1

The second portion asks the participants to describe in detail *your thought on the quality of the video, quality of mouth movements, how the original compares to the generated ones, and your overall feelings about the synthetic avatar*. The third portion asked the participants to compare the model using the child avatar and the model using the Obama avatar according to some criteria, see table 4.3, in which the participants are asked to select the best result. The fourth portion asks the participants to describe *Why did you prefer one avatar above the other one?*. The fifth portion asks the participants *what would be the most important aspect that should be improved for this type*. And at last, the participants are asked about their age group and gender.

Q	Question from third portion
Q1	I found that the following avatar was more realistic than the other one.
Q2	I found that the following avatar had the more realistic mouth movement compared to the other
Q3	Overall I liked the following avatar better compared to the other.

Table 4.3: Question from third portion

4.1.3 Results and Discussions

The participants seemed to prefer Obama avatar over the child avatar, this was overall due to the child avatar shivering more than the Obama avatar. While method 2 reduced the quality disparity between the avatars, the Obama avatar was still significantly more preferred. We will go into greater depth about the results, beginning with question portions one and two for each avatar and finishing with a comparison section for question portions three, four, and five.

Full answers for question portion one and two is available in the appendix, see appendix A.1.

Obama avatar: The participants appeared to favor the second method in general, see table 4.4. The second method was described as causing less shivering and preserving higher video quality than the first method but having the disadvantage of the mouth seeming quite closed for extended portions of the video. Due to the effect of the second method making the mouth appear more closed, the participants gave method 1 a higher rating on Q3; *The voice appears convincingly to come from the person*. The shivering was described as the main cause of quality loss in the first method, which could describe why the participants preferred the second method. Overall, the methods trades off between quality and a higher range of motion in the mouth region.

Question	method 1 (mean)	method 2 (mean)
Q1	2.4	3
Q2	2.4	2.8
Q3	2.2	2.6

Table 4.4: Answer for questions regarding Obama avatar for table 4.2

Child Avatar: The first method was described to be more jittering than the second method. Overall, both methods were described to contain too much shivering, making the video look unrealistic. The child avatar mouth movement was also described to deviate from the expected mouth moment regarding the input audio. Overall the participants preferred method 2, where Q1 and Q2 scored higher while Q3 was scored less, see table 4.5.

Question	method 1 (mean)	method 2 (mean)
Q1	1.6	2.4
Q2	1.8	2.4
Q3	2.2	2

Table 4.5: Answer for questions regarding child avatar for table 4.2

Comparison of Obama and Child avatar

When comparing the child avatar to the Obama avatar for questions from the third portion table 4.3, the consensus was that the Obama avatar was preferable, see table 4.6. The participants agreed more on Q1 regarding realism, while Q2 was more even, but the participants still slightly preferred the Obama avatar.

Child vs Obama	Child	Obama
Q1	1	4
Q2	2	3
Q3	1	4

Table 4.6: Scores for child vs Obama for questions from table 4.3

The reason for this is primarily due to the Obama avatar's reduced shivering around the mouth area. Even though the participants favored the Obama avatar, they deemed the child avatar's realism to be comparable to that of the Obama avatar. The participants also stated the mouth-movement from the Obama avatar seemed more realistic, the child avatars lip movement was stated to bear no resemblance to the source mouth movement, see table 4.7.

Preferred	Why did you prefer one avatar above the other one? Please describe.
Obama	The realism in the mouth-movement
Obama	I feel that the both having the same quality. I do not have that option above
Obama	Easier to see that the child mouth is generated compared to the Obama
Child	It was more realistic
Obama	The vibration around the mouth in the Obama video is much less than the child video. Also, in the child video, the lip movements bear no resemblance to the desired alphabet. And only the child's body movement is normal of because it is taken from the original video

Table 4.7: Which avatar did the participants prefer and why? Question portion 4.

The majority of participants (3/5) believed that the most important aspect

that could be improved was the reduction of jittering/shivering around the mouth area (see table 4.8), which we suspect to be caused mostly by estimation error in the face detector and feature extractor. We should explore methods for reducing estimation errors in conjunction with a more sophisticated smoothing strategy to account for estimation error. Two of the participants suggested that we expand the model to include the whole face, which could easily be performed by not cropping the 3D face mesh, but we purposefully limited the model to the mouth area by cropping the 3D face mesh. This was done in order to simplify and narrow the thesis's subject.

What would be the most important aspect that should be improved for this type of avatars?
I think that the movements of the mouth in both should be a bit more/ a bit more opening of the mouth while talking (as we see the researcher is)
Rather than focusing on only mouth, if the whole face can be generated with facial expression, it may help to improve the overall quality
A more visually smooth synthetic mouth. This makes the "fake" aspect of the video more discoverable compared a detailed mouth movements following the actual sound
that the movements would be stabilized and the sound is synchronized with the movement
Reduce extra vibrations and move the lips according to the said sentence, not be immobile. In more advanced adapt the body, eye and hand movement according to the speech

Table 4.8: Feed back from the participants regarding improvement

4.2 Quantitative evaluation

Using the feedback from the participant, we establish some questions regarding the performance difference between the Obama avatar and the Child avatar; *is the performance difference due to the difference in data set quality?*

We can investigate the question using a couple of different methods; firstly, we can compare the L2 values used when selecting the n best candidate frames. It is fair to assume that a shorter L2 distance between the source and the target expression allows the model to select more appropriate candidate frames,

resulting in a better match between the synthesized mouth region and the desired mouth region.

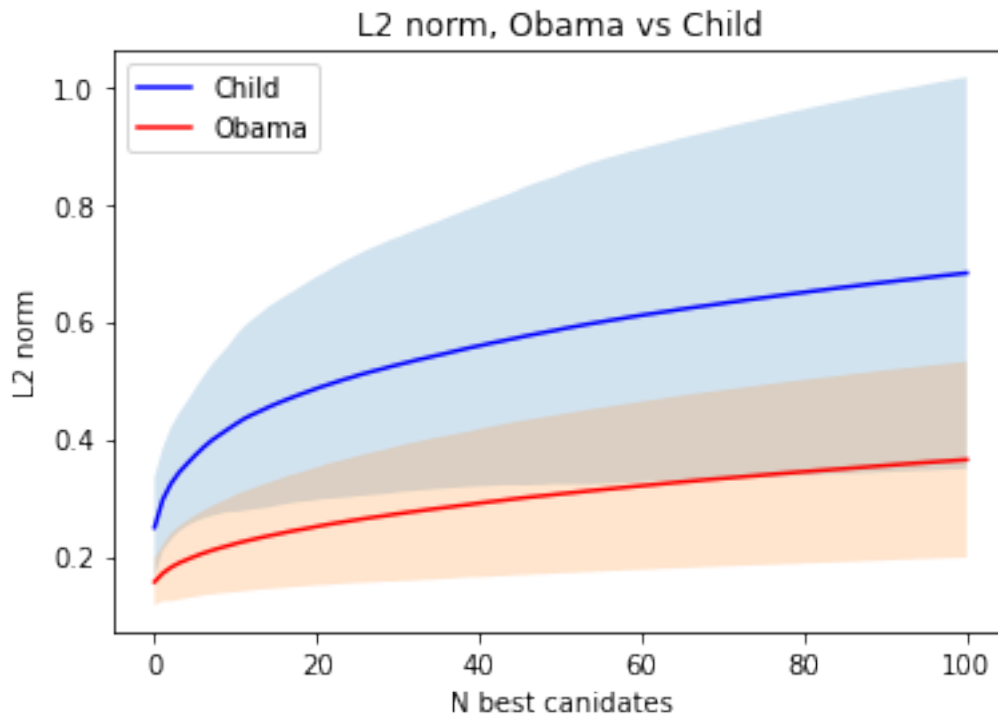


Figure 4.2: Average L2-distance between source and N-closest target avatar expression, lower is better. The lightly colored area indicated the standard deviation

We can see that the Obama avatar appears to find better candidate frames, which could account for the difference in performance, see fig 4.2. This could be related to the source's facial shape being more similar to the Obama avatar; the L2 distance between the average α_{source} and α_{Obama} is equal to 2.06×10^5 (sum pixel distance between vertices), while the L2 distance between α_{source} and α_{child} is equal to 2.87×10^5 . We can further examine if the difference in quality is attributable to a shape difference between the source and target avatars by utilizing a source face similar to the target avatar, used for confirming our question. If the L2 distance between the child avatar and the Obama avatar is similar when utilizing α_{source} equal α_{target} , we may conclude that the discrepancy is not attributable to differences in training data.

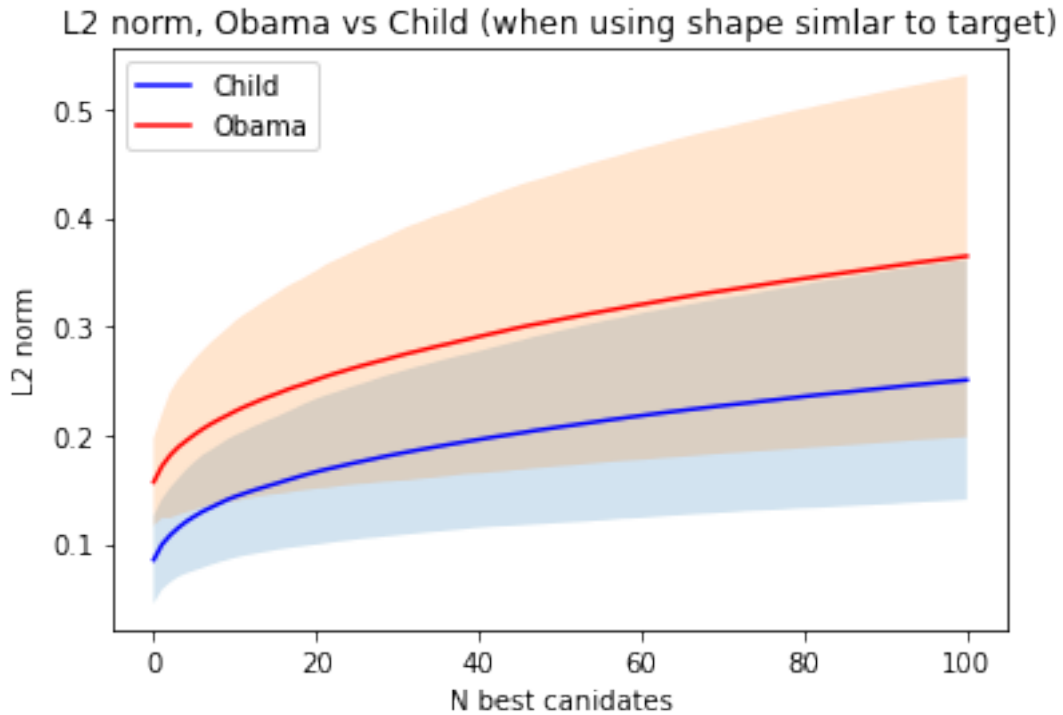


Figure 4.3: Average L2-distance between source and N-closet target avatar expression, where source identity is equal target identity

When using α_{source} equal α_{target} , the $L2$ distance decreased significantly, notably for the child avatar, which even outperformed the Obama avatar, see fig 4.3. This should imply that the child avatar does not have inferior training data (collected frames used to select n best frames) and that the differences between the α_{source} and α_{target} have a significant role when it comes to the quality of the generated avatar. Note that the 3DDFA model and the BFM model might contribute as well to the quality difference between the Child avatar and Obama avatar; both models (3DDFA and BFM) seem to have been trained on a well-distributed data set. However, biases are hard to account for especially assuming the face shape of a child differs significantly from the mean population. Recent study's on commercial and open-source facial recognition algorithms shows a negative bias for each algorithm on children [34], which could explain why the Obama avatar performs better.

4.2.1 Quantitative evaluation when source avatar is equal target avatar

We can evaluate the model’s reconstruction ability by using the same video as both the source and the target so that the input image I_i^S equals the target image I_i^T . We will be using the both avatar for this section, where the first 250 frames are denoted as our source and target video, while the remaining frames are used to compute the estimated lower facial texture, such the model has no prior knowledge of True frame. This allows for deriving meaningful reconstruction error, since we now know the true value, unlike if we had used a source video different from the target video. Since we have the true value, we can evaluate the reconstructed images against the original/true image. Note that this evaluation does not cover the cases where the source and target identities are different, and does not account for when the rendered lower face shape does not match the target face, see section 4.3 for information regarding the mismatch limitation. We will evaluate both the temporal and spatial preforms individually.

Spatial evaluation:

For the spatial evaluation; we will be using the metrics SSIM and LPIPS for evaluation the perceptual quality loss and NME for the facial landmark displacement. We are using FAN [7] for estimating the facial landmarks.

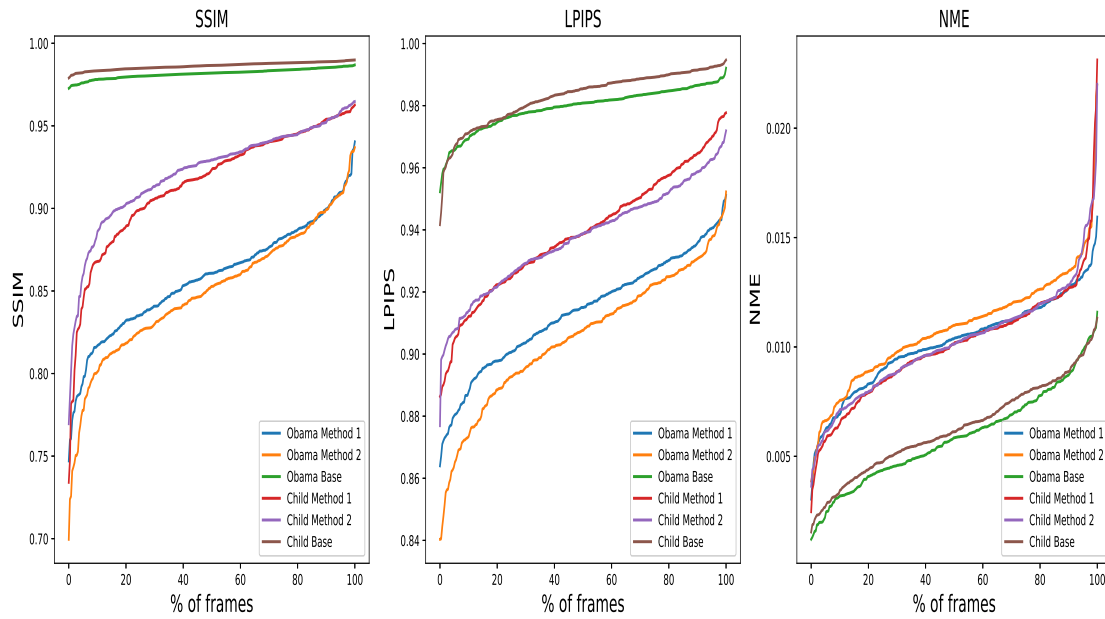


Figure 4.4: SSIM, LPIPS and NME for both avatars, methods and base loss, for both for SSIM and LPIPS lower y-value is better while higher is better for NME

The base loss L_{base} is defined for when the model knows the true vertex colors, which we use as a reference point, where L_{base} estimates the reconstruction loss caused by the feature extractor, renderer, and blender.

The child avatar does significantly better for the both perceptual losses, which corresponds with findings in previews section; the $L2$ distance between source and n best candidate frames is significantly better for the child avatar. For NME, the result do not change significantly, see fig 4.4.

Temporal evaluation:

For temporal evaluation; we will be using the metrics tLP and tOF for evaluating the temporal quality losses.

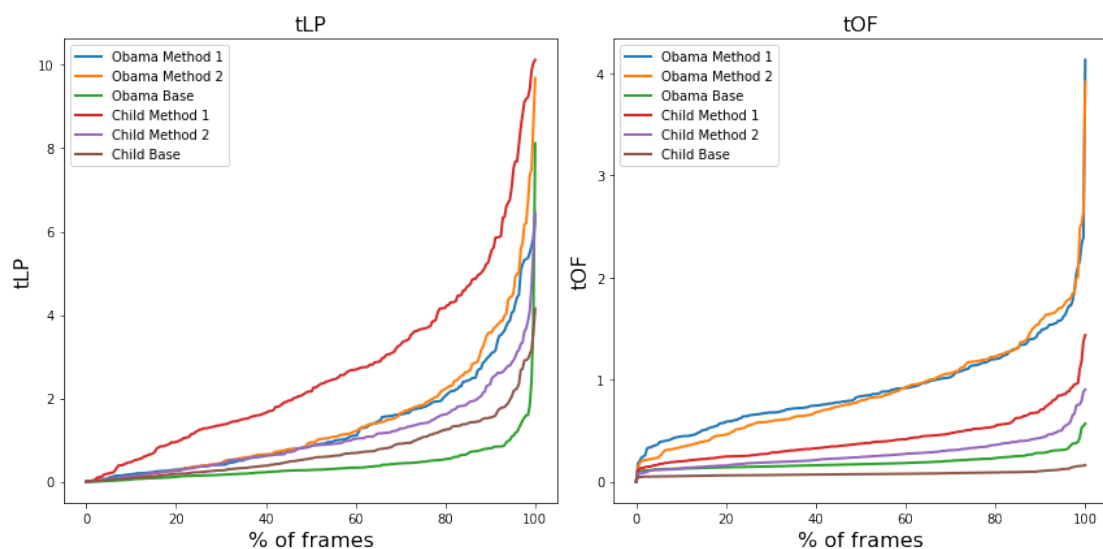


Figure 4.5: tLP and tOF for both avatars, methods and base loss, for both metrics lower y-value is better

We can see a clear improvement from method 1 too method 2 for the tLP metrics, especially for the child avatar. For both models the child avatar preforms considerably better, see fig 4.5.

4.2.2 Evaluation of n best candidates

Once the loss L_{base} is established, we can compute the vertex-color using n best candidate frames to estimate the loss caused by the facial texture synthesizer by simply subtracting the new reconstruction loss $L_{base+fs}$ by the L_{base} , resulting in $L_{fs} = L_{base+fs} - L_{base}$. In addition, by varying n when selecting n best frames, we can investigate how the number of n candidate frames affects the image's quality. We will be using SSIM for this evaluation.

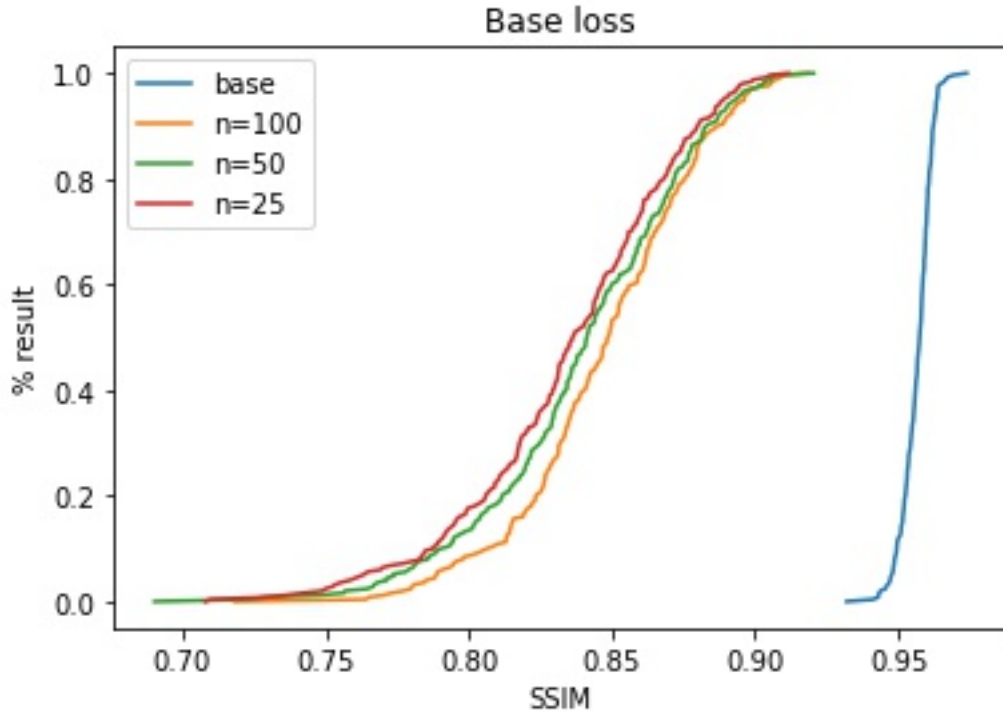


Figure 4.6: SSIM value for different n best candidate values. Base is when true vertex-colors are known

The model seems to lose substantial quality when estimating the vertex color, and the number of n best candidates seems to not amount to a big overall improvement. The most significant spatial error in the system seems to be introduced in the facial texture method.

4.3 Limitations

When we blend the rendered lower face with the target face, the rendered face might not fit. This is due to the difference between the target expression and the source expression being too large. The lower face vertices are reconstructed by passing α_{target} and β_{source} as input to eq. (2.6), which reconstructs vertices of the target identity deformed by the source expression; if this deformation does not match the true target shape, we get an unnatural blend of the rendered lower face and target face. For example, suppose the target avatar has his mouth open,

and we try to retarget the avatar with a closed mouth. In that case, we get a target avatar with a closed mouth, and an unnaturally extended chin see fig 4.7. The synthetic lower face will not fit because the chin is extracted/lifted when the mouth is closed. This may not appear odd in a single image, but it appears quite unnatural over a sequence where the mouth shape does not change while the chin fluctuates.

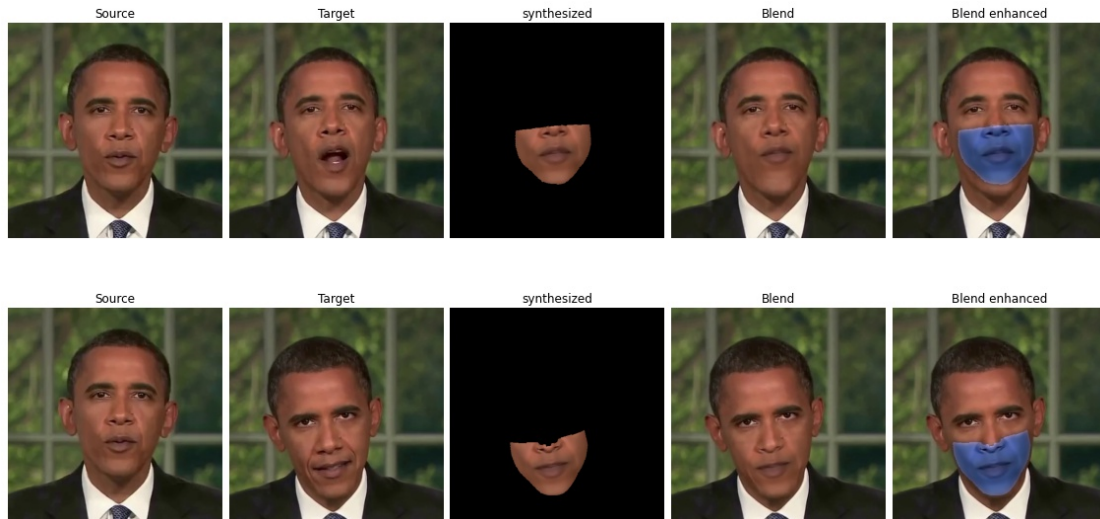


Figure 4.7: The synthesized image is computed using the source image, and the blend image is a blend of the synthesized and target image. Row 1 is denotes a bad fit between the synthesized image, while Row 2 to is good fit. The blue color is the overlapped parts form synthesized and target. Note how Obama’s chin in blend row 1 look unnaturally long.

4.4 System resources

The experiment is run on a Laptop with; Windoes system, NVIDIA Geforce GTX 1060 GPU and Intel(R) Core(TM) i7-8750H CPU. The model uses on average 160ms per frame, further breakdown of run time for each non-negligible (time wise) method is given in the table below 4.9. The timeit² module is used to measuring the run time. The timeit module provides a simple

²timeit: <https://docs.python.org/3/library/timeit.html>

way to time small bits of Python code. The module automatically determines the appropriate number runs and returns the average time measured to execute the selected code.

Method	time (ms)
Face detector	34
Feature extractor	20
Weighted median	41
Renderer	25
Blender	30

Table 4.9: Time measurement for non-negligible time consuming methods

Nota bene, the measurement does not account for additional time consumed by capturing and streaming if applied in real-time. The model could be further optimized; both the face detector and feature extractor are reported to run significantly faster in the original papers and implementation; the face detector is stated to run at 120 fps on a GPU, while the feature extractor runs at 2ms when using ONNX runtime³. ONNX runtime is a cross-platform, high performance machine learning inferencing and training accelerator. Even with those further improvements, the model would still operate at a maximum of 10-11 frames per second depending on how fast the computer can read a video stream of the source actor, which would still feel lagging considering that most videos run at speeds greater than 30 frames per second. If we wish to improve the run performance further, we would need to upgrade the hardware or reimplement the blender, renderer, and weighted median method.

4.5 Summary

This section compared the model using two distinct data sets, evaluated it under both ideal and not ideal conditions (aside from possibly a better data set), and discussed some of the model's drawbacks. The model appeared to perform significantly better when the source person resembled the target person, owing

³<https://github.com/microsoft/onnxruntime>

to the model discovering better matches in the collected frames (training set). The model's largest source of spatial noise was introduced by the lower facial texture estimator, as expected, given that the lower facial texture estimator is the primary component.

Chapter 5

Summary and Conclusions

The task of facial reenactment is a complex and tedious task, which have been studied extensively. There are several methods to approach this task, a common approach used in SOTA models such as [14][39] usually consists of a 3DMM in conjunction with a deep neural network. More specific a layout of using an deep neural network such as a CNN to parameterize a 3DMM with target identity and source expression, then use another deep neural network to synthesize an texture for the modified 3d face representation and at last blending the synthesize face into target avatar.

Inspire by this approach and the synthesize Obama paper[35], we attempted to combine the two in order to accomplish our thesis's initial objective; *Using an actor to reenact the expression form actor to a target avatar*. Rather than utilizing a neural network to synthesize the facial texture, we adapted the weighted median method described in [35]. [35] took advantage of the weighted median's quality-preserving properties to synthesize a photorealistic facial texture by computing the weighted median of the n most similar images to the target image. The distance between the source mouth key points and a collection of the targets mouth key points was used to determine the n most similar images. Rather than computing a weighted median for each pixel in the synthesized mouth region, we attempted to sample the pixels by representing the mouth region as a dense alignment of points with their associated vertex-colors using a 3DMM. By leveraging the 3DMM, we were able to sample

in a deterministic manner, guaranteeing that each image in our collection was represented consistently. Thus, we synthesize our facial texture by first, computing the weighted median of the vertex-colors in the n most similar images as defined by the distance between the actor's and n target person's expressions in our collection. And secondly, using the computed vertex colors and their associated vertices to render the mouth region using rasterization. The rendered mouth region is then blended to the target avatar using Laplacian pyramid blend, resulting in a target avatar with the expression of the source target. We further try to improve our model by incorporating a smoothing method to reduce temporal inconsistency introduced by our texture estimation and 3d face parameterization.

We evaluate our model with and without smoothing on two distinct avatars using a questionnaire, which assists us in achieving our second objective; *Compare the performance of our model on two distinct avatars and examining the effect of the dataset on the model's performance.* We test our model on an adult avatar and child avatar to examine how our model is affected by the dataset used for synthesizing the avatar and determine if the used 3DMM and facial feature extractor performed as well on the child avatar as the adult avatar. The questionnaire feedback indicated that the adult avatar outperformed the child avatar significantly. To investigate why this appeared to be the case, we examined the quality of the n similar images used to synthesize the facial texture and discovered that our model was impacted by the distribution difference between source and target expression. When the source face was more similar to the target face, our model appeared to locate higher-quality n similar images, we define quality by computing L2 distance between source and target expression. Additionally, we analyzed the model when the target identity was equal to the source identity; in this case, the child avatar performed better, indicating that the model performed equally well on a child avatar as it did on an adult avatar; the performance difference in the questionnaire was due to the source identity being more similar to the target avatar identity.

5.1 Main contribution

The research completed throughout the thesis aims to answer the problem statement assigned in section 1.2. Our main contribution given those defined objectives are as follows:

Objective 1: *Research whether we can make the visual avatar’s facial expression follow the audio by reenacting the expression associated with the source actor’s audio.*

We developed a model that can reenact lower facial expressions from an actor to a target avatar. The model followed the actor’s audio to a certain extent, but the results appeared to be dependent on the difference between the actor’s and avatar’s expression characteristics, with a big disparity resulting in a poor reenactment. Taking this into account, we created a model that synthesizes photo-realistic lower facial texture for the avatar; the result look good when displayed singly, but was significantly worse when utilized to generate video sequences.

Objective 2: *Comparing the performance of our model on two distinct avatars and examining the effect of the dataset on the model performance.*

We discovered that the adult avatar performs better than the child avatar using a questionnaire; this could be because the pre-trained models are more representative of adult faces, implying that the data sets used to train the feature extractors contain more adult samples. We examined the model further and determined that the performance difference was caused by the source actor’s identity being more close to the Obama avatar identity. When the source identity was identical to the target identity, the child avatar performed similarly to or better than the Obama avatar.

5.2 Future work

While the model is capable of reenacting expressions from actor to target avatar, additional work can be done to make the output appear more realistic. To further develop the model, we would like to address the limitations discussed

in Section 4.3 and incorporate comments from survey participants, as shown in tables 4.7 and 4.8. By doing so, we may separate the key areas for future improvement into two categories: reduced shivering and more realistic blending.

Reduced shivering: The shivering is caused by multiple parts; noise introduced when parameterizing the actor and bad quality of n best candidates for the facial texture synthesizing part. Firstly, we can reduce the noise introduced when parameterizing the actor by implementing the noise reduction method described in [28] (section 3.2). This is done by re-initialize the original bounding box n times by perturbing it by few pixels length in various directions of the image plane, and then averaging the resulting sets of parameters, effectively reducing the variance of the 3FFDA result by increasing sample size. Secondly, we can improve the quality of n best candidates by using the expression transfer method from [38], which transforms the actor expression to the space of target expression. By applying this method, we can find better n best candidate matches because the actor and target distributions are more aligned

More realistic blending: We can address the blending limitations that result in the target expression not matching the target shape semantically, i.e., closed mouth should have lower chin than open mouth. Because our current blending method doesn't take into account the target expression being correlated with the target face shape, by eroding the chin region in the target avatar we can use image inpainting to fill the eroded space to better fit with the new target expression. This is a regularly used strategy for resolving comparable issues, as demonstrated in [39].

Bibliography

- [1] URL: <https://cs231n.github.io/convolutional-networks/>.
- [2] URL: https://www.uio.no/studier/emner/matnat/its/nedlagte-emner/UNIK4690/v17/forelesninger/lecture_2_3_blending.pdf.
- [3] Joyce A Adams, Karen J Farst and Nancy D Kellogg. 'Interpretation of medical findings in suspected child sexual abuse: an update for 2018'. In: *Journal of pediatric and adolescent gynecology* 31.3 (2018), pp. 225–231.
- [4] Ethem Alpaydin. *Introduction to machine learning*. MIT press, 2020.
- [5] Gunn Astrid Baugerud et al. 'Multimodal Virtual Avatars for Investigative Interviews with Children'. In: *Proceedings of the 2021 ACM Workshop on Intelligent Cross-Data Analysis and Retrieval*. 2021, pp. 2–8. DOI: 10.1145/3463944.3469269. URL: <https://doi.org/10.1145/3463944.3469269>.
- [6] Deirdre A Brown and Michael E Lamb. 'Forks in the road, routes chosen, and journeys that beckon: A selective review of scholarship on children's testimony'. In: *Applied Cognitive Psychology* 33.4 (2019), pp. 480–488.
- [7] Adrian Bulat and Georgios Tzimiropoulos. 'How far are we from solving the 2d & 3d face alignment problem?(and a dataset of 230,000 3d facial landmarks)'. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2017, pp. 1021–1030.
- [8] Chen Cao et al. 'Facewarehouse: A 3d facial expression database for visual computing'. In: *IEEE Transactions on Visualization and Computer Graphics* 20.3 (2013), pp. 413–425.
- [9] Ann-Christin Cederborg et al. 'Investigative interviews of child witnesses in Sweden'. In: *Child abuse & neglect* 24.10 (2000), pp. 1355–1361.

- [10] François Chollet. 'Xception: Deep learning with depthwise separable convolutions'. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 1251–1258.
- [11] Mengyu Chu et al. 'Temporally coherent gans for video super-resolution (tecogan)'. In: *arXiv preprint arXiv:1811.09393* 1.2 (2018), p. 3.
- [12] Jia Deng et al. 'Imagenet: A large-scale hierarchical image database'. In: *2009 IEEE conference on computer vision and pattern recognition*. Ieee. 2009, pp. 248–255.
- [13] Peter J. Denning et al. 'Computing as a discipline'. In: *Computer* 22.2 (1989), pp. 63–70.
- [14] Michail Christos Doukas, Stefanos Zafeiriou and Viktoriia Sharmanska. 'HeadGAN: One-shot Neural Head Synthesis and Editing'. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 14398–14407.
- [15] Bernhard Egger et al. '3d morphable face models—past, present, and future'. In: *ACM Transactions on Graphics (TOG)* 39.5 (2020), pp. 1–38.
- [16] Gunnar Farneback. 'Two-frame motion estimation based on polynomial expansion'. In: *Scandinavian conference on Image analysis*. Springer. 2003, pp. 363–370.
- [17] Ronald Aylmer Fisher et al. '012: A Mathematical Examination of the Methods of Determining the Accuracy of an Observation by the Mean Error, and by the Mean Square Error.' In: (1920).
- [18] Ian J. Goodfellow et al. *Generative Adversarial Networks*. 2014. arXiv: 1406.2661 [stat.ML].
- [19] Jianzhu Guo et al. 'Towards Fast, Accurate and Stable 3D Dense Face Alignment'. In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2020.
- [20] Yudong Guo et al. 'AD-NeRF: Audio Driven Neural Radiance Fields for Talking Head Synthesis'. In: *IEEE/CVF International Conference on Computer Vision (ICCV)*. 2021.

- [21] Berthold KP Horn and Brian G Schunck. 'Determining optical flow'. In: *Artificial intelligence* 17.1-3 (1981), pp. 185–203.
- [22] Andrew G Howard et al. 'Mobilenets: Efficient convolutional neural networks for mobile vision applications'. In: *arXiv preprint arXiv:1704.04861* (2017).
- [23] Miriam Johnson et al. 'Best practice recommendations still fail to result in action: A national 10-year follow-up study of investigative interviews in CSA cases'. In: *Applied Cognitive Psychology* 29.5 (2015), pp. 661–668.
- [24] Ian T Jolliffe. 'Principal components in regression analysis'. In: *Principal component analysis* (2002), pp. 167–198.
- [25] Niels Krause et al. 'The effects of feedback and reflection on the questioning style of untrained interviewers in simulated child sexual abuse interviews'. In: *Applied Cognitive Psychology* 31.2 (2017), pp. 187–198.
- [26] Michael E Lamb et al. *Tell me what happened: Questioning children about abuse*. John Wiley & Sons, 2018.
- [27] Leo Liberti et al. 'Euclidean distance geometry and applications'. In: *SIAM review* 56.1 (2014), pp. 3–69.
- [28] Jacek Naruniec et al. 'High-resolution neural face swapping for visual effects'. In: *Computer Graphics Forum*. Vol. 39. 4. Wiley Online Library. 2020, pp. 173–184.
- [29] *A 3D Face Model for Pose and Illumination Invariant Face Recognition*. IEEE. Genova, Italy, 2009.
- [30] Francesco Pompèdda, Angelo Zappalà and Pekka Santtila. 'Simulations of child sexual abuse interviews using avatars paired with feedback improves interview quality'. In: *Psychology, Crime & Law* 21.1 (2015), pp. 28–52.
- [31] Sebastian Ruder. 'An overview of gradient descent optimization algorithms'. In: *arXiv preprint arXiv:1609.04747* (2016).

- [32] Karen Simonyan and Andrew Zisserman. *Very Deep Convolutional Networks for Large-Scale Image Recognition*. 2015. arXiv: 1409.1556 [cs.CV].
- [33] Linsen Song et al. *Everybody's Talkin': Let Me Talk as You Want*. 2020. arXiv: 2001.05201 [cs.CV].
- [34] Nisha Srinivas et al. 'Face Recognition Algorithm Bias: Performance Differences on Images of Children and Adults'. In: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. 2019, pp. 2269–2277. DOI: 10.1109/CVPRW.2019.00280.
- [35] Supasorn Suwajanakorn, Steven M Seitz and Ira Kemelmacher-Shlizerman. 'Synthesizing obama: learning lip sync from audio'. In: *ACM Transactions on Graphics (ToG)* 36.4 (2017), pp. 1–13.
- [36] Richard Szeliski. *Computer vision: algorithms and applications*. Springer Science & Business Media, 2010. Chap. 3.5, pp. 167–185.
- [37] John Tabak. *Geometry: the language of space and form*. Infobase Publishing, 2014, p. 150.
- [38] Justus Thies et al. 'Face2face: Real-time face capture and reenactment of rgb videos'. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 2387–2395.
- [39] Justus Thies et al. 'Neural voice puppetry: Audio-driven facial reenactment'. In: *European Conference on Computer Vision*. Springer. 2020, pp. 716–731.
- [40] Zhou Wang et al. 'Image quality assessment: from error visibility to structural similarity'. In: *IEEE transactions on image processing* 13.4 (2004), pp. 600–612.
- [41] Eric W Weisstein. 'Convolution'. In: <https://mathworld.wolfram.com/> (2003).
- [42] Richard Zhang et al. 'The Unreasonable Effectiveness of Deep Features as a Perceptual Metric'. In: *CVPR*. 2018.
- [43] Shifeng Zhang et al. 'Faceboxes: A CPU Real-time Face Detector with High Accuracy'. In: *IJCB*. 2017.

Appendix A

Google Forms Questionnaire

In this appendix, we present the Google Forms questionnaire used for the subjective study, and the respective results not displayed in chapter 4.

A.1 Answers

Table A.1: Answers to Obama questions, fig A.1 and A.2

Method 1			Method 2		
Q1	Q2	Q3	Q1	Q2	Q3
Good	Agree	Neutral	Good	Agree	Agree
Poor	Disagree	Disagree	Fair	Neutral	Disagree
Poor	Disagree	Disagree	Fair	Neutral	Disagree
Poor	Disagree	Strongly disagree	Poor	Disagree	Disagree
Poor	Disagree	Neutral	Fair	Disagree	Neutral

Table A.2: Answers to child questions, fig A.1 and A.2

Method 1			Method 2		
Q1	Q2	Q3	Q1	Q2	Q3
Poor	Disagree	Neutral	Poor	Neutral	Neutral
Poor	Disagree	Disagree	Poor	Disagree	Disagree
Bad	Strongly Disagree	Disagree	Fair	Disagree	Disagree
Poor	Disagree	Disagree	Fair	Neutral	Disagree
Bad	Strongly disagree	Disagree	Poor	Disagree	Strongly Disagree

Please watch the video presenting all three methods in parallel plus the original one and try to describe in detail your thought on quality of the video, quality of mouth movements, how the original compares to the generated ones and your overall feelings about the synthetic avatar?(Obama)

I think the overall quality of the video/picture was 4 , but the mouth and the degree of movement in the eyes are challenging. The mouth in the first generated one was moving more naturally I think, but not so much in sync with the researcher. The mouth in the advanced (second) was somehow very much closed and in the start the whole lower part of the face/mouth and "hake" moved. I could wish for some more movement while pronouncing the letters or trying with a sentence? Eye movement was more natural I think according to the blinking. The movement of the eye (eye gaze?) was somehow missing, but that might not have much to say as long as the closing/ blinking is natural.

A more visually smooth synthetic mouth. This makes the "fake" aspect of the video more discoverable compared a detailed mouth movements following the actual sound

You do see that the mouth is not the original. It is also a bit hard to see if the generated video actually follows the audio. I think the generated video has potential, but there is a job do wrt. making it more smooth.

The second method shows better video quality compared to the first one. It would be nice to test the same sequence with bigger time gap.

Blinking eyes and moving the head look very natural. But the vibration around the mouth and especially the chin is so weak.

Table A.3: Answer for fig. A.3

<p>Please watch the video presenting all three methods in parallel plus the original one and try to describe in detail your thought on quality of the video, quality of mouth movements, how the original compares to the generated ones and your overall feelings about the synthetic avatar?(Obama)</p>
<p>I think the lower part of the face - the mouth and chin, destroy how I perceive this one. Looking at the original, I see that the boy moves his mouth to a lesser extent but still, more than in the transfer examples. Watching it now, I didn't see too much difference between the first and the advanced - got too occupied looking at the mouth-area I think. The eye-movement I think was much the same quality as the Obama-one, I just didn't catch it that well being distracted by the mouth and chin. Looking at this one, I see that the researcher actually moves his mouth more when saying the letters than I first thought. My suggestion then might fall to the ground (sentences..).</p>
<p>Improving mouth movement between pauses can improve the overall quality.</p>
<p>Again, for the generated video, there is easy to see that the mouth is changed from the original. It is even more visible here than the Obama videos. The mouth movement itself seems to be better here, but it is more visible that it is not the original mouth. The generated videos have potentials, but some more smoothing is needed here too.</p>
<p>Similar to the Obama video - the mouth is sort of shivering which makes the video look unrealistic. However, in this video the movement of the mouth is more realistic.</p>
<p>The vibration around the mouth is high, especially in the first video. Also, does not change his mouth while saying many alphabets in both videos.</p>

Table A.4: Answer for fig. A.6

A.2 Questions


Questionnaire used to evaluate and compare a child avatar to an adult avatar. The questionnaire was conducted using Google forms and includes questions about the quality of the reenacted avatars.

Obama Method 1

In this study we will present you two videos from a face and mimic transfer algorithm.

paalh@simula.no (not shared) [Switch account](#)

Method 1



How would you rate the overall video quality?

- Bad
- Poor
- Fair
- Good
- Excellent

The visual appearance of the video is realistic.

- Strongly disagree
- Disagree
- Neutral
- Agree
- Strongly agree

The voice appears convincingly to be coming from the person in the video.

- Strongly disagree
- Disagree
- Neutral
- Agree
- Strongly agree


[Next](#) [Clear form](#)

Figure A.1: Video of reenacted Obama avatar using method 1, and questions from question portion 1, see table 4.2. User is asked to score the method.

Obama Method 2

In this study we will present you two videos from a face and mimic transfer algorithm.

Method 2



How would you rate the overall video quality?

- Bad
- Poor
- Fair
- Good
- Excellent

The visual appearance of the video is realistic.

- Strongly disagree
- Disagree
- Neutral
- Agree
- Strongly agree

The voice appears convincingly to be coming from the person in the video.


- Strongly disagree
- Disagree
- Neutral
- Agree
- Strongly agree

[Back](#) [Next](#) [Clear form](#)


Figure A.2: Video of reenacted Obama avatar using method 2, and questions from question portion 1, see table 4.2. User is asked to score the method regarding given topic.

Obama comparison

Overall



Original video Obama



Please watch the video presenting all three methods in parallel plus the original one and try to describe in detail your thought on quality of the video, quality of mouth movements, how the original compares to the generated ones and your overall feelings about the synthetic avatar. *

Your answer


Back Next Clear form

Figure A.3: Side by side comparison of method 1 and 2 for Obama avatar, with the original source actor for reference and a separate video of the original target video. User is asked to describe thoughts about overall feel and quality of reenacted avatars

Child Method 1

In this study we will present you two videos from a face and mimic transfer algorithm.

Method 1



How would you rate the overall video quality?

- Bad
- Poor
- Fair
- Good
- Excellent

The visual appearance of the video is realistic.

- Strongly disagree
- Disagree
- Neutral
- Agree
- Strongly agree

The voice appears convincingly to be coming from the person in the video.

- Strongly disagree
- Disagree
- Neutral
- Agree
- Strongly agree

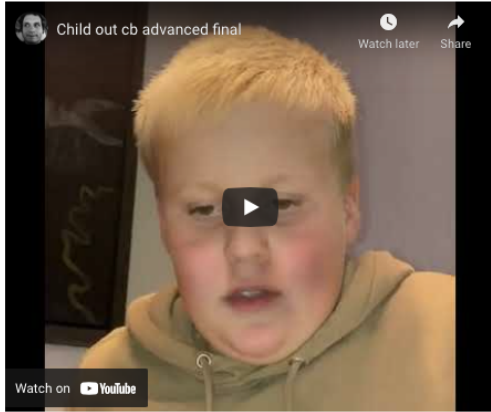
[Back](#) [Next](#) [Clear form](#)

Figure A.4: Video of reenacted child avatar using method 1, and questions from question portion 1, see table 4.2. User is asked to score the method regarding given topics.

Child Method 2

In this study we will present you two videos from a face and mimic transfer algorithm.

Method 1



How would you rate the overall video quality?

- Bad
- Poor
- Fair
- Good
- Excellent

The visual appearance of the video is realistic.

- Strongly disagree
- Disagree
- Neutral
- Agree
- Strongly agree

The voice appears convincingly to be coming from the person in the video.


- Strongly disagree
- Disagree
- Neutral
- Agree
- Strongly agree

[Back](#) [Next](#) [Clear form](#)


Figure A.5: Video of reenacted child avatar using method 2, and questions from question portion 1, see table 4.2. User is asked to score the method regarding given topics.

Child comparison

Overall



Original video child



Please watch the video presenting all three methods in parallel plus the original one and try to describe in detail your thought on quality of the video, quality of mouth movements, how the original compares to the generated ones and your overall feelings about the synthetic avatars. *

Your answer

[Back](#) [Next](#) [Clear form](#)

Figure A.6: Side by side comparison of method 1 and 2 for child avatar, with the original source actor for reference and a separate video of the original target video. User is asked to describe thoughts about overall feel and quality of reenacted avatars.

Final questions

I found that the following avatar was more realistic than the other one.

Obama
 Child

I found that the following avatar had the more realistic mouth movement compared to the other.

Obama
 Child

Overall I liked the following avatar better compared to the other.

Obama
 Child

Why did you prefer one avatar above the other one? Please describe.

Your answer _____

What would be the most important aspect that should be improved for this type of avatars?

Your answer _____

What is your gender?

Your answer _____

What is your age group?

10-20
 20-30
 30-40
 40-50
 51 and more

[Back](#) [Submit](#) [Clear form](#)

Figure A.7: Comparison of child and Obama avatar, user is asked to choose the preferred avatar for given questions, see table 4.3. User is also encouraged to give feedback on aspects to improve and state gender and age group