Skoltech

Skolkovo Institute of Science and Technology

MASTER'S THESIS

Facial attribute editing using neural networks

Master's Educational Program: Data Science

Student: _____

Evgeny Zholkovskiy
Data Science
June, 2021

Research Advisor: _____

Ivan Oseledets
Professor

Moscow 2021

Skoltech

Skolkovo Institute of Science and Technology

МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ

Редактирование черт лица с использованием нейронных сетей

Магистерская образовательная программа: Наука о данных

Студент:_____
Евгений Жолковский
Наука о данных
Июнь, 2021

Научный руководитель:_____
Оселедец Иван Валерьевич
Профессор

Москва 2021

Facial attribute editing using neural networks

Evgeny Zholkovskiy

Submitted to the Skolkovo Institute of Science and Technology
on June, 2021

Abstract

In recent years, with a huge boost of neural networks and artificial intelligence, many computer vision methods and algorithms became widely used in computational photography, image and video editing, entertainment, and medicine. These methods allow to accelerate and automate image editing and help with many tasks.

Of particular interest are methods that can automatically edit images with given conditions and properties for different tasks. These include image colorization and styling, removing and fixing defects on a photo, adding and editing objects, and other tasks. Such models are growing in popularity in image editing programs and services. It is especially interesting and challenging to adopt these methods for facial photos because each face has many small details that are unique for every person. Also, such editing should produce a photo-realistic result because even a tiny facial image defect strongly affects human perception.

The subject of this work is an implementation of a particular case of facial editing - removing braces from a photo of a person. A comprehensive multistage image processing pipeline was implemented to solve this task. Also, a big dataset for the braces removing task was collected. A quantitative evaluation of the proposed pipeline was performed using crowdsourcing.

Research Advisor:
Name: Ivan Oseledets
Degree: PhD
Title: Professor

Редактирование черт лица с использованием нейронных сетей

Евгений Жолковский

Представлено в Сколковский институт науки и технологий
Июнь, 2021

Реферат

В последние годы активно развиваются нейронные сети и искусственный интеллект, и множество методов и алгоритмов компьютерного зрения на их основе стали широко использоваться в вычислительной фотографии, редактировании изображений и видео, сфере развлечений и медицине. Эти методы позволяют ускорить и автоматизировать редактирование изображений во многих задачах. Интересны методы, которые могут производить автоматическое редактирование изображений с заданными свойствами. Это могут быть задачи колоризации и стилизации изображений, а также исправления или удаления дефектов, добавления и редактирования объектов на фотографии. Такие методы становятся все более популярны в программах и сервисах для редактирования изображений.

Особенный интерес представляет собой применение этих методов для фотографий лица, потому что оно имеет много мелких деталей и особенностей, уникальных для каждого человека. Кроме того, такое редактирование должно иметь очень хорошее качество, потому что даже небольшой дефект на фотографии лица сильно влияет на его восприятие человеком.

Темой данной работы является реализация частного случая редактирования лица - удаление брекетов с фотографии. Для решения этой задачи был реализован многошаговый алгоритм для обработки изображений. Также был собран большой набор изображений для обучения модели. Количественная оценка предложенного алгоритма была проведена с использованием краудсорсинга.

Научный руководитель:
Имя: Оселедец Иван Валерьевич
Ученое звание, степень: Д-р физ.-мат. наук
Должность: Профессор

Acknowledgments

I want to thank Ulyana Pihlak for industry-related problem statement, prof. Ivan Oseledets and Daniil Polyakov for help with experiments, a lot of helpful discussions, and ideas.

Contents

List of Figures

Chapter 1

Problem Statement

The subject of this work is an implementation of an algorithm that removes dental braces from a photo of a person. This algorithm should keep the resolution of an initial image and produce a photo-realistic result that can be shared in social networks.

Removing braces from a photo of a person is a particular case of facial attribute editing. This problem was explored in many works [4, 15, 22, 1]. These approaches exploit generative adversarial networks to produce realistic results. But generative models encounter typical problems. Editing a small part of an image affects the whole image and may produce artifacts. Also, it takes a lot of time and computing resources to train a network that works with high-resolution photos. As well it's difficult and expensive to obtain a high-quality training dataset with examples for different domains.

An image processing pipeline is proposed in this work for the braces removal task. This pipeline tackles the problems described above and allows to utilize advantages and power of generative neural networks.

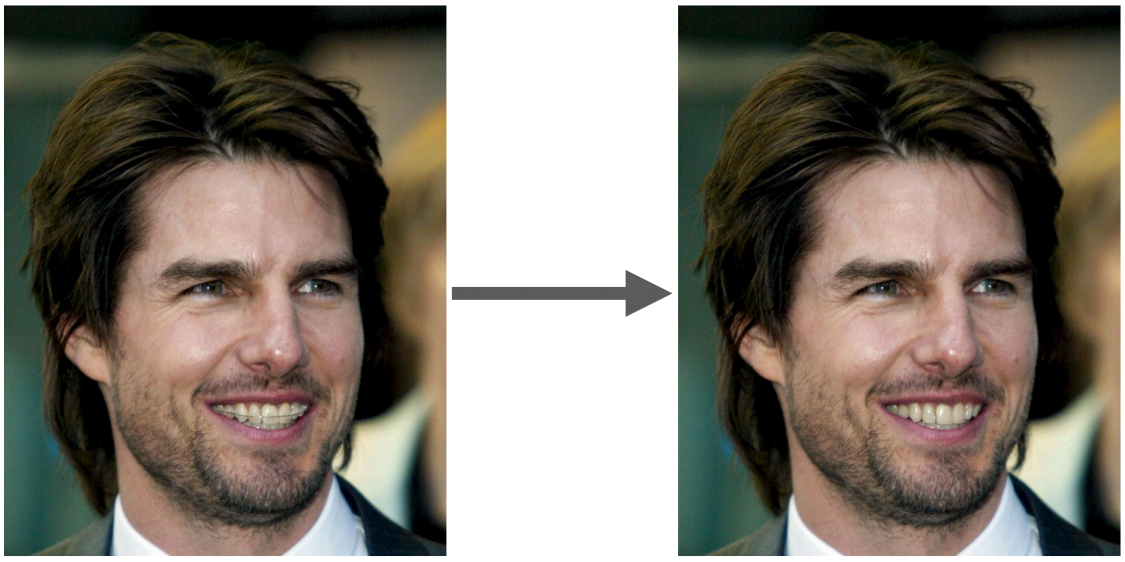

Figure 1.1: Example of result that should be achieved.

Chapter 2

Image Processing Pipeline

The image processing pipeline consists of several steps. To perform an accurate image to image transition and image restoration precise estimation of mouth position is required. The idea is to find all persons on a photo, crop each face, and estimate mouth position and shape. This two stages approach makes mouth estimation more robust and lets it work in various illumination and semantic conditions.

2.1 Face detection

Multi-task Cascaded Convolutional Networks (MTCNN) architecture proposed in [28] is used for face detection on images. This method is widely used for face detection tasks and has a lot of implementations.

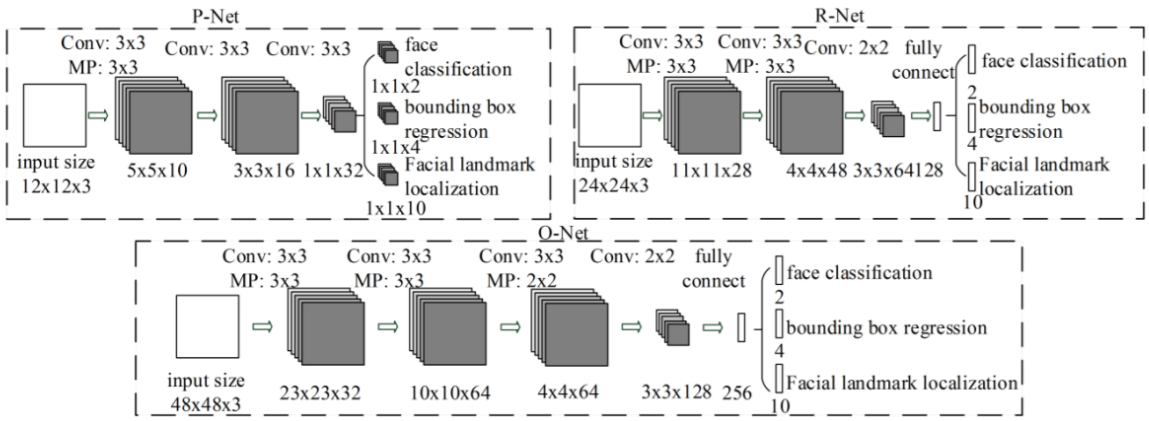

Figure 2.1: MTCNN architecture.

The architecture is a cascaded neural network with 3 stages for detecting and aligning faces and facial landmarks localization. It is presented on 2.1. An input image is resized to several resolutions to build an image pyramid. In the first stage Proposal Network (P-Net) proposes candidate bounding boxes and then non-maximum suppression is applied. The next stage is called Refine Network (R-Net). It rejects many bounding boxes and calibrates proposal bounding boxes. The last stage is Output Network (O-Net). This stage returns the final bounding box positions and also returns 5 landmarks for each face. During training three losses are used for the face detection prob-

lem. Cross-entropy loss is used to classify if the predicted bounding box matches the ground truth box.

$$\mathcal{L}_i^{det} = -(y_i^{det} \log p_i + (1 - y_i^{det})(1 - \log p_i))$$

For estimating location of bounding box and face landmarks mean squared error loss is used.

$$\mathcal{L}_i^{box} = \|\hat{y}_i^{box} - y_i^{box}\|^2$$

$$\mathcal{L}_i^{landmark} = \|\hat{y}_i^{landmark} - y_i^{landmark}\|^2$$

Though this architecture has face landmarks detection, this is not enough for the task because it is not possible to estimate mouth shape using only this information. So another model is needed for mouth aligning.

2.2 Mouth align

2.2.1 Face landmarks

Face landmark detection task is a task of detecting key landmarks of a face and tracking them. There are several datasets with labeled keypoints on photos such as 300 Faces In-the-Wild [18] and Menpo dataset [6, 27, 26]. In both of these datasets, each face has 68 keypoints (some of keypoints may be occluded). These keypoints give sufficient shape approximation for cropping and aligning mouth. Examples of images with facial landmarks are presented in figure 2.2.

Figure 2.2: Examples of image with facial landmarks from Menpo dataset.

Resnet-18 [9] network is used for this task. This is a deep convolutional neural network with residual connections between layers. These connections are used to prevent gradient vanishing during training. A model with 136 outputs (2 coordinates for each of 68 keypoints) was trained to minimize mean squared error loss for the location of each keypont. Geometrical augmentations such as scaling and cropping only part of a face to make the model robust. This approach shows good results when a face is not occluded, but it is not robust when only part of the face is visible or the face has a strange expression. Examples of a situation when the model fails to correctly detect landmarks are presented in figure 2.3. For solving this problem semantic segmentation approach is used.

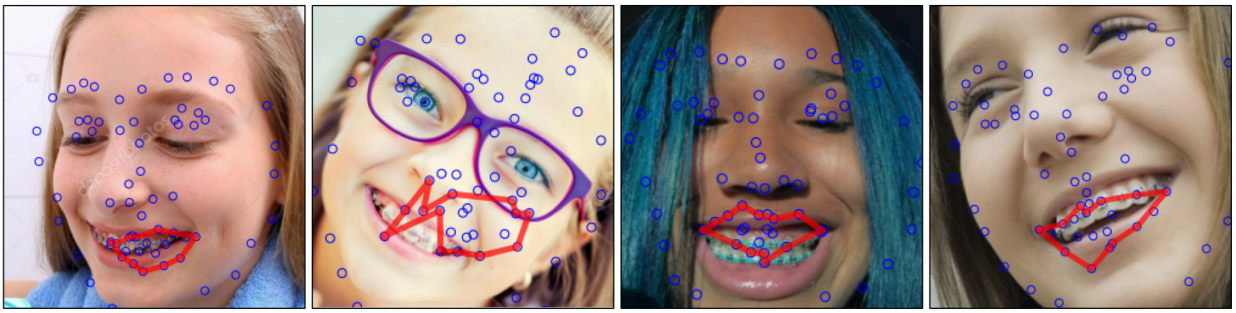

Figure 2.3: Examples of fails of the facial landmarks model.

2.2.2 Semantic Segmentation

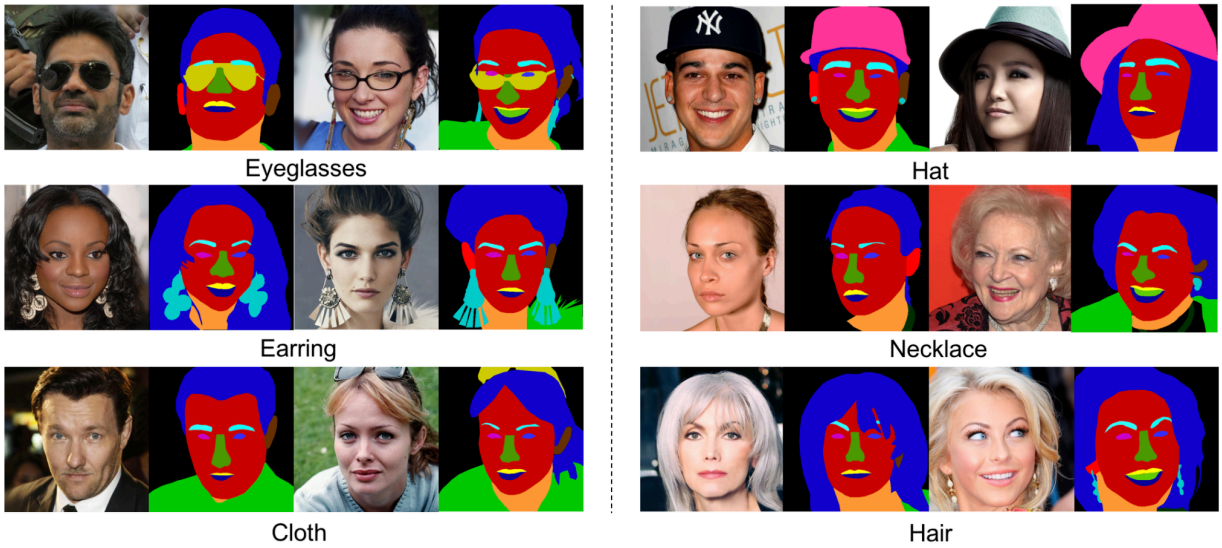

Figure 2.4: CelebAMask HQ dataset examples.

Semantic Segmentation is the task of classifying each pixel of an image into one of several classes. Face semantic segmentation is more robust for the task of aligning mouth because it can detect mouth even when only part of a face is visible. CelebAMask HQ Dataset [15] dataset is used for training the segmentation model. Examples of this dataset are shown in figure 2.4.

This dataset contains 30 thousand high-resolution face images with pixelwise annotation of 19 classes such as nose, eyes, lips, and tooth. Bilateral Segmentation Network (BiSe) [25] is used for semantic segmentation of a face. A schematic representation of this network is presented on 2.5. The main idea of this architecture is to use spatial and the context paths for inference. The spacial path preserves high resolution for spatial information but the receptive field is not sufficient. The context path is designed to provide a large receptive field. It uses Xception architecture [5] as a backbone features extractor to encode high-level semantic information. Also, context path has 2 intermediate outputs which are fused with the path outputs using the feature fusion module. Cross entropy loss is used for training. The over loss is a combination of a principal loss for the final output X and auxiliary losses for intermediate outputs X_k that are used to stabilize model optimization during training.

$$\mathcal{L}_{ce} = -\sum_{i} \log(\frac{\exp p_i}{\sum_j \exp p_j})$$

$$\mathcal{L}_{total} = \mathcal{L}_{ce}(X) + \alpha \sum_{k} \mathcal{L}_{ce}(X_k)$$

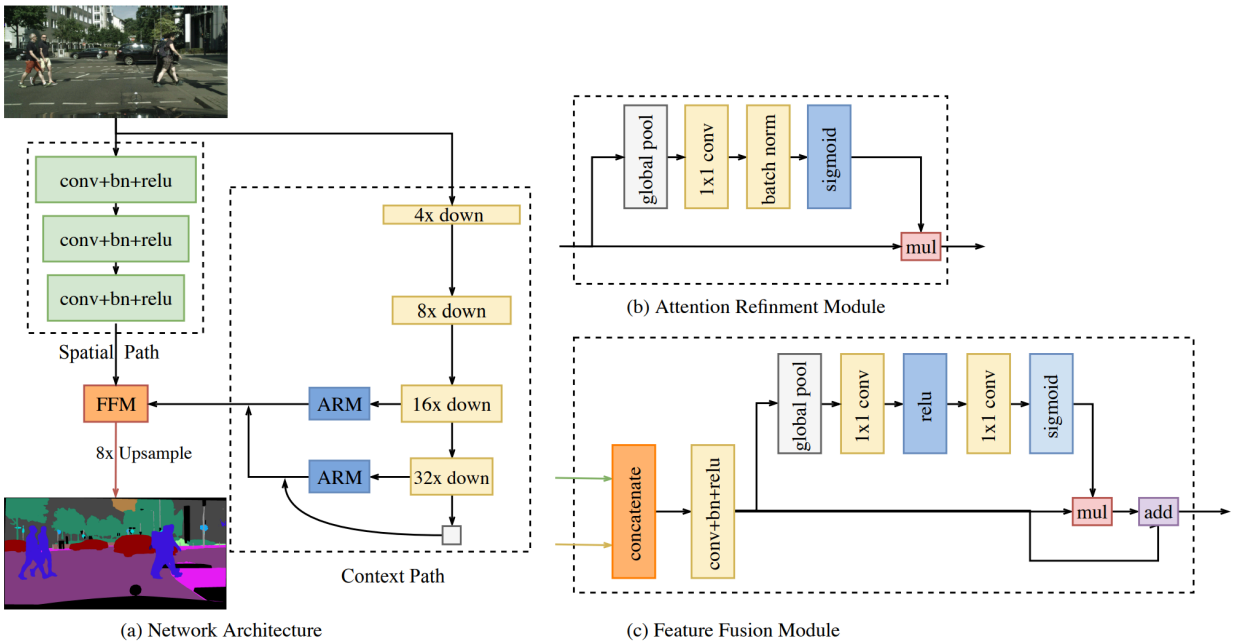

Figure 2.5: Bilateral Segmentation Network architecture.

Mouth segmentation is more robust than face landmarks detection and shows good results even if only part of a face is visible. To increase overall pipeline recall segmentation model is applied even if MTCNN detector fails to detect a face. This may happen because the photo was taken too close to a face and has no background. Examples of such situations are presented in figure 2.6.

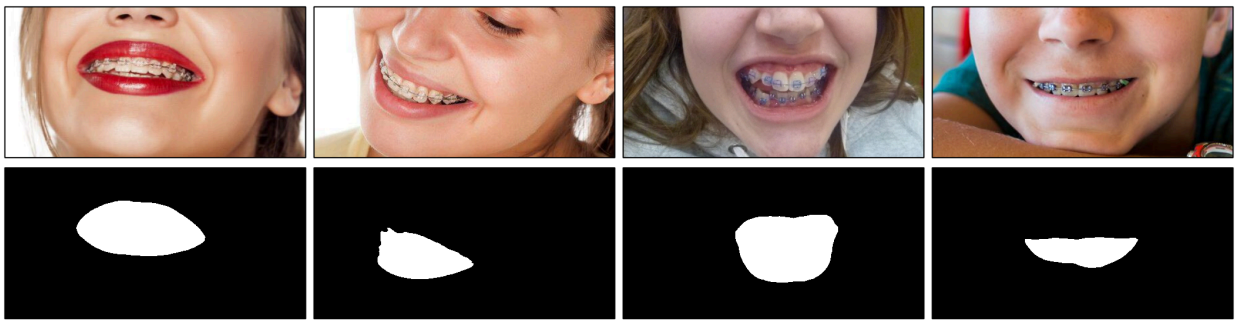

Figure 2.6: Examples of segmentation results when face detector fails to detect face.

2.2.3 Mouth alignment

The largest area which is classified as lips or teeth is considered as a mouth mask to prevent shape artifacts. A convex hull is calculated for this area and considered as a mouth shape. The hull is approximated with an ellipse. A mouth bounding box with a fixed ratio is calculated using this ellipse location, orientation, and size. This procedure is shown on the pipeline scheme in figure 2.15

2.3 Braces removing

Two approaches can be used to obtain a photo-realistic and consistent image of the same mouth without braces.

2.3.1 Image Inpainting

The first approach has two stages. In the first stage, segmentation is applied to find all pixels corresponding to braces that need to be removed. In the second stage, these pixels are filled using inpaining algorithm. Image inpainting is a task of realistic filling in holes in an image. There are various image inpaining algorithms such as Partial Convolutions [16], EdgeConnect [17], Deep image prior [21]. Examples of inpainting algorithm results are presented in figure 2.7.

Both of these stages need special datasets: segmentation needs a large number of images with pixelwise braces labeling and inpainting model needs a dataset of mouth images without braces with realistically generated braces shape mask to perform well. There is no publicly available dataset for this task. These datasets can be obtained using crowdsourcing but it would take a lot of time and be very expensive.

Figure 2.7: Examples of image inpainting algorithm results. Partial Convolutions on the left. EdgeConnect on the right.

2.3.2 Generative Models

Another way to remove braces is to use generative adversarial neural networks (GANs). Generative networks presented in [8] became an important and powerful tool in computer vision tasks for generating photo-realistic images with specific properties. Pix2pix [11] framework uses generative networks to learn a mapping from input to output images. For training cross-domain image transition using Pix2pix paired examples from both domains are required. CycleGAN framework [29] tackles this problem and allows to train models that can learn mapping from one domain to another with unpaired data. CycleGAN principle architecture is presented on 2.8.

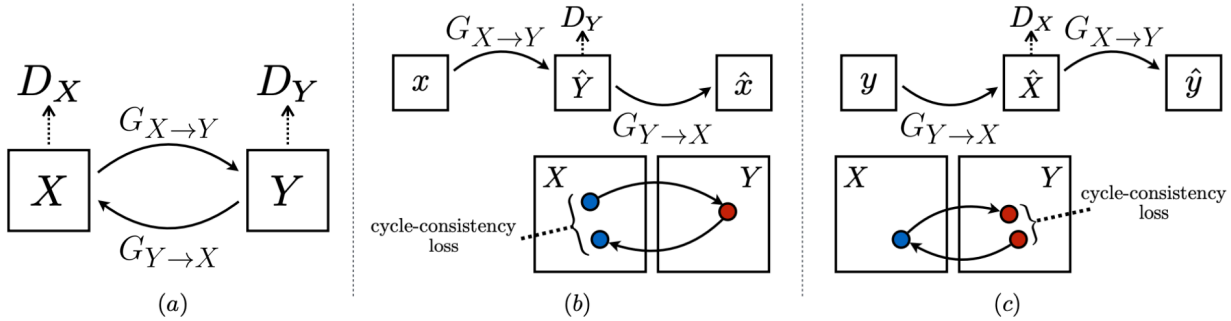

Figure 2.8: CycleGAN principle architecture.

The idea is to train two generators: $G_{X\to Y}$ to perform a transition from domain X to domain Y and $G_{Y\to X}$ to perform a transition from Y to X. Also, two discriminators D_X and D_Y are trained. D_X aims to distinguish between images from domain X and images generated using $G_{Y\to X}$ and the same for D_Y. Training objective contains two types of losses: adversarial loss and cycle consistency loss. Adversarial loss is used for matching the generated images distribution to target

the domain data distribution.

$$\mathcal{L}_{GAN}(G_{X\to Y}, D_Y) = \mathbb{E}_{y\in Y}[\log D_Y(y)] + \mathbb{E}_{x\in X}[1 - \log D_Y(G_{X\to Y}(x))]$$

$$\mathcal{L}_{GAN}(G_{Y\to X}, D_X) = \mathbb{E}_{x\in X}[\log D_X(x)] + \mathbb{E}_{y\in Y}[1 - \log D_X(G_{Y\to X}(y))]$$

Using generators $G_{X\to Y}$ and $G_{Y\to X}$ described above an image x from domain X can be transited to domain Y and then back to domain X. The same can be done with an image y from domain Y.

$$x \to G_{X\to Y}(x) \to G_{Y\to X}(G_{X\to Y}(x)) \approx x$$

$$y \to G_{Y\to X}(y) \to G_{X\to Y}(G_{Y\to X}(y)) \approx y$$

After these transitions result should be similar to the input image. To make training more stable and reduce space of possible mapping functions consistency loss $\mathcal{L}_{cons}$ is used:

$$\mathcal{L}_{cycle}(G_{X\to Y}, G_{Y\to X}) = \mathbb{E}_{x\in X}[\|G_{Y\to X}(G_{X\to Y}(x)) - x\|]_1 + \mathbb{E}_{y\in Y}[\|G_{X\to Y}(G_{Y\to X}(y)) - y\|]_1$$

The total loss is

$$\begin{aligned}\mathcal{L}_{total}(G_{X\to Y}, G_{Y\to X}, D_X, D_Y) = \\ \mathcal{L}_{GAN}(G_{X\to Y}, D_Y) + \mathcal{L}_{GAN}(G_{Y\to X}, D_X) \\ +\lambda_{cycle}\mathcal{L}_{cycle}(G_{X\to Y}, G_{Y\to X})\end{aligned}$$

CycleGAN cross-domain image transitions examples are presented in figure 2.9. Such results are achieved using only unpaired datasets of images for each domain. It can be seen that domain transition using this framework also changes the background color for some examples which is crucial for braces removing task. Therefore it is necessary to use more modern and complex models for image-to-image transition [4, 20, 24, 10] that also use the approach that was presented in CycleGAN.

For the braces removing task, a big dataset with source and target domains for the braces removing task was obtained. The source domain consists of 14 thousand aligned and cropped images of mouth without braces and the target domain contains 1.5 thousand images with braces. Examples from this dataset are presented in figure 3.2. Details of obtaining this dataset are described in chapter 3.

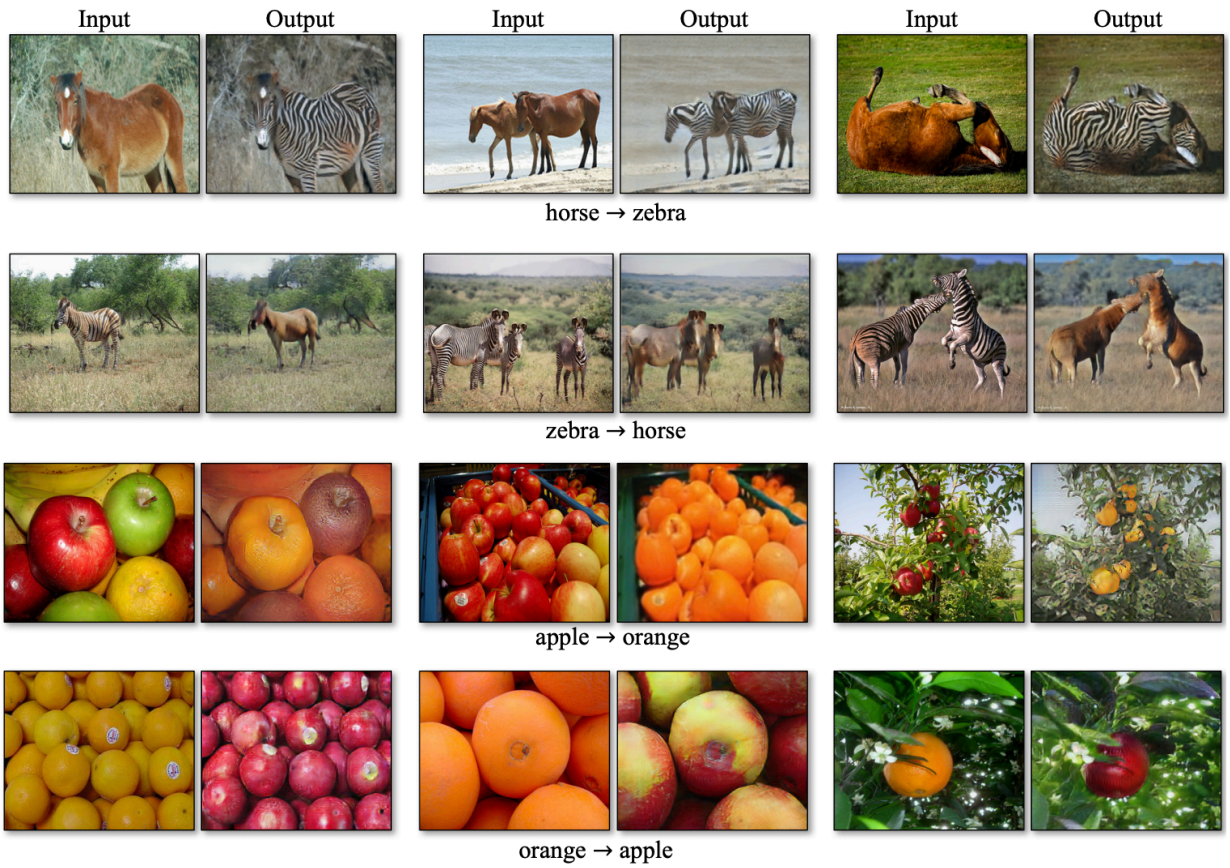

Figure 2.9: CycleGAN cross-domain image transition results.

2.3.3 Attention-Guided GAN

Attention-Guided Generative Adversarial Networks (AGGAN) for Unsupervised Image-to-Image Translation [20] (AGGAN) is a framework that evolves CycleGAN approach for cross-domain image transition. AGGAN is designed to detect the most discriminative semantic parts of an image using an attention mechanism and minimize changes and artifacts in other parts of the image. This framework is used in the work because these properties are crucial for face editing tasks.

A schematic representation of AGGAN architecture is presented in figure 2.10. This architecture also uses cycle and adversarial $\mathcal{L}_{GAN}$ and cycle $\mathcal{L}_{cycle}$ losses, which were described above. AGGAN generators architecture is a modified architecture from [12] with a built-in attention mechanism. Each generator has 11.8 million parameters which is comparable with ResNet-18 network that has 11.7 million parameters. Both generators have three outputs for an input:

$$G_{X \to Y} : x \to [M_y, R_y, G_y]$$

$$G_{Y \to X} : y \to [M_x, R_x, G_x]$$

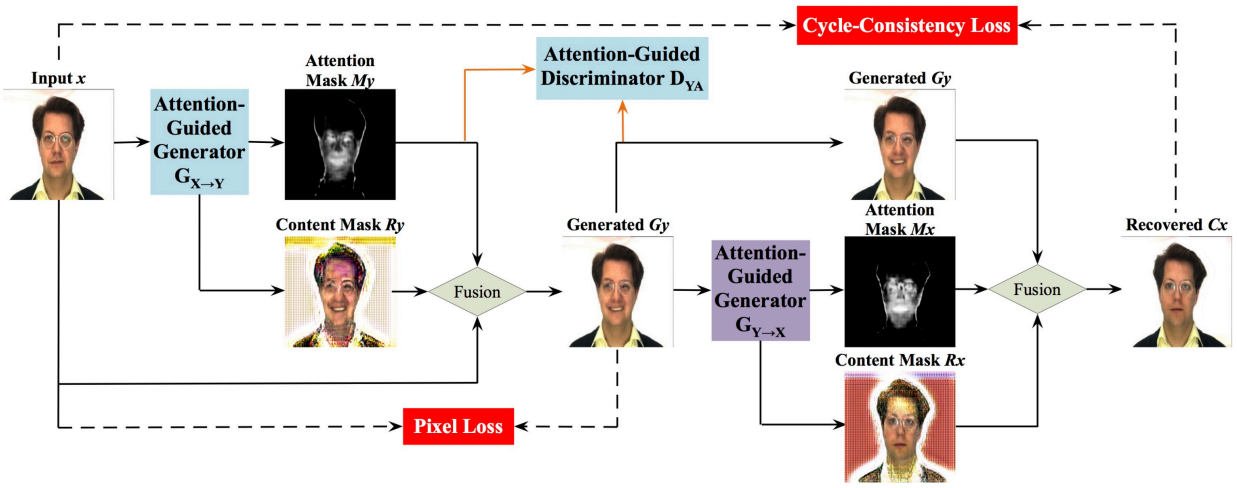

Figure 2.10: Principal scheme of Attention-guided GAN.

Here M_x, M_y are attention masks which indicate regions that should be affected in output, R_x, R_y are content masks with generated content, and G_x, G_y are generator final results. They are calculated using following formulations:

$$G_x = R_x * M_x + y * (1 - M_x)$$

$$G_y = R_y * M_y + x * (1 - M_y)$$

AGGAN proposes a new attention-guided adversarial loss:

$$\mathcal{L}_{AGAN}(G_{X\to Y}, D_{YA}) = \mathbb{E}_{y\in Y}\left[\log D_{YA}([M_y, y])\right] + \mathbb{E}_{x\in X}\left[\log(1 - D_{YA}([M_y, G_{X\to Y}(x)]))\right]$$

$$\mathcal{L}_{AGAN}(G_{Y\to X}, D_{XA}) = \mathbb{E}_{x\in X}\left[\log D_{XA}([M_x, x])\right] + \mathbb{E}_{y\in Y}\left[\log(1 - D_{XA}([M_x, G_{Y\to X}(y)]))\right]$$

D_{YA} is a discriminator that distinguishes generated images G_y that were obtained using a learned attention mask and real images $y \in Y$. Same for D_{XA}

Attention masks are not trained explicitly because there are no ground truth masks for braces. Masks are learned from discriminator gradients. Attention mask can easily saturate to 1 which eliminates the advantage of the attention-guided generator. To prevent this problem a total variation regularization is applied over masks. Total variation loss is presented below:

$$\mathcal{L}_{tv}(M_i) = \sum_{w,h=1}^{W,H} |M_i(w+1,h) - M_i(w,h)| + |M_i(w,h+1) - M_i(w,h)|\,, i \in [x, y]$$

Pixel loss is used to reduce changes between input and generated images to prevent artifacts and constraints generator. This loss can make the result worse in case if source and target domains are very different, but in the case of braces removing only a small part of an image should be

changed. Pixel loss is expressed as:

$$\mathcal{L}_{pixel}(G_{X\to Y}, G_{Y\to X}) = \mathbb{E}_{x\in X}[\|G_{Y\to X}(y) - x\|]_1 + \mathbb{E}_{y\in Y}[\|G_{X\to Y}(x) - y\|]_1$$

Identity loss is not described in AGGAN paper [20], but it is implemented in a code provided by the authors. This loss reduces changes between input and generated image in case if target domain if the input image is from the target domain. Thereby the model will only edit images with braces and images without braces won't change.

$$\mathcal{L}_{identity}(G_{X\to Y}, G_{Y\to X}) = \mathbb{E}_{x\in X}[\|G_{Y\to X}(x) - x\|]_1 + \mathbb{E}_{y\in Y}[\|G_{X\to Y}(y) - y\|]_1$$

The overall loss is formulated as:

$$\begin{aligned}
&\mathcal{L}_{total}(G_{X\to Y}, G_{Y\to X}, D_X, D_Y) = \\
&\quad \lambda_{GAN} * [\mathcal{L}_{GAN}(G_{X\to Y}, D_Y) + \mathcal{L}_{GAN}(G_{Y\to X}, D_X)] \\
&+\lambda_{AGAN} * [\mathcal{L}_{GAN}(G_{X\to YA}, D_{YA}) + \mathcal{L}_{GAN}(G_{Y\to X}, D_{XA})] \\
&+\lambda_{cycle} * \mathcal{L}_{cycle}(G_{X\to Y}, G_{Y\to X}) \\
&+\lambda_{pixel} * \mathcal{L}_{pixel}(G_{X\to Y}, G_{Y\to X}) \\
&+\lambda_{identity} * \mathcal{L}_{identity}(G_{X\to Y}, G_{Y\to X}) \\
&+\lambda_{tv} * [\mathcal{L}_{tv}(M_x) + \mathcal{L}_{tv}(M_y)]
\end{aligned}$$

Figure 2.11: Examples of braces removing using AGGAN.

The network accepts images of size 384x192. This resolution is sufficient to keep small details of teeth and demonstrate good results even for high-resolution inputs. The following training strategy was used: the model was trained for 40 epochs with an initial learning rate and 10 epochs with the learning rate linearly descending to 0. Adam optimizer [14] was used for training. Examples of images with removed braces are shown in figure 2.11.

2.3.4 Colour correction

Generative neural networks show good results in image-to-image transition tasks, but they tend to change image colours and tones. Examples of this problem for the braces removing task are presented in figure 2.12. Solving this problem using additional losses during training can significantly affect the result. Therefore post-processing for color correction can be more preferable.

Figure 2.12: Examples of color tone change for generated images.

A schematic representation of a color correction algorithm is presented in figure 2.13. Attention-guided GAN that is used for removing braces produces attention mask M_x. The joint mask is a combination of the attention mask M_x and a mouth mask obtained using the segmentation model. The joint mask is used to select a set of pixels without braces on the input image P_{input} and corresponding pixels on the generated output P_{output}.

A polynomial transform for color correction [3] where P_{output} is a source domain and P_{input} is a target domain is learned for these pixels and applied to the generated image. The polynomial transform with 5 terms that are used in the image processing pipeline is presented below:

$$\begin{cases} R' = \alpha_1 R + \beta_1 G + \gamma_1 B + \delta_1 RGB + \epsilon_1 \\ G' = \alpha_2 R + \beta_2 G + \gamma_2 B + \delta_2 RGB + \epsilon_2 \\ B' = \alpha_3 R + \beta_3 G + \gamma_3 B + \delta_3 RGB + \epsilon_3 \end{cases}$$

where $(R, G, B) \in P_{output}$ are pixels from the generated image and $(R', G', B') \in P_{input}$ are pixels

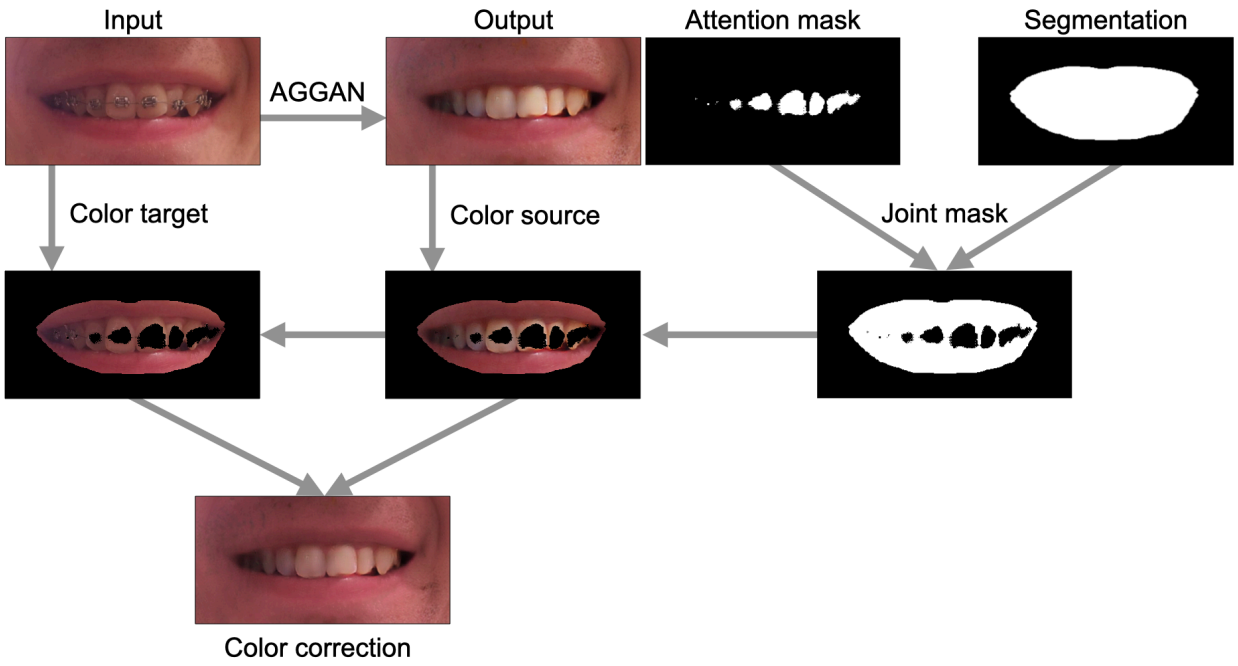

Figure 2.13: Schematic representation of the color correction algorithm.

from the input image with braces. The least-square solution of the system is obtained using the Moore-Penrose inverse.

Examples of the results of the color correction results are presented in figure 2.14. For some images, skin color around the mouth changes but this part of the image is removed when the results are placed back on the original image. After the colour correction, the result is projected back on the original image using mouth and face coordinates and segmentation masks that were obtained in the pipeline.

Figure 2.14: Examples of the color correction results.

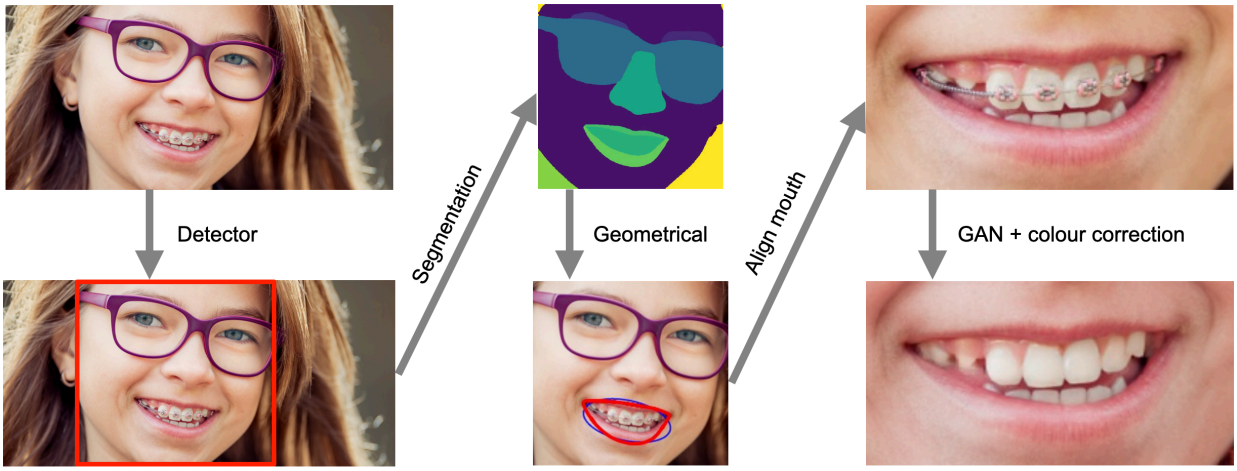

Figure 2.15: Schematic representation of the image processing pipeline.

2.4 Inference time

A schematic representation of the developed image processing pipeline is presented in figure 2.15. Some of the operations are implemented using neural networks and can be run on GPU devices and may have different inference times. Measurements were conducted on an image with a resolution of 1024x1024 with a single face on it. Each operation was run multiple times to obtain better statistics. Mean values of duration for all pipeline operations are presented in table 2.4.

Table 2.1: Inference time for operations in the image processing pipeline.

	Core i7, 2.6 GHz	GeForce RTX 2080 Ti
Face detection	204 ms	138 ms
Face segmentation	670 ms	65 ms
Mouth alignment	165 ms	
GAN	1070 ms	23 ms
Colour correction	18 ms	
Wrap result	31 ms	
Total	2158 ms	440 ms

Chapter 3

Data preparation

3.1 Dataset parsing

A large two domains dataset is required to train a generative network. Two sources of images were used to obtain the dataset. A Flickr-Faces-HQ Dataset (FFHQ) [13] was used to build a domain of people without braces. This dataset consists of 70 thousand high-resolution photos of persons of various ages, ethnicity in different environments and lighting conditions. These images were filtered using the image processing pipeline described in chapter 2. A two-stage semi-automatics cleaning process was used to remove images where the teeth are not visible. At the first stage, the face semantic segmentation model was applied and all images without predicted teeth mask are dropped. In the second stage, Resnet-18 classification model was used to increase dataset precision further. This model accepts a cropped image of a mouth and returns the probability that teeth are visible on the image. The model was trained on manually labeled images with and without visible teeth. Each class had about one thousand labeled images. About 14 thousand images of faces without braces were obtained using the described cleaning procedure

The second domain consisting of photos of people with braces was obtained using image retrieval system parsing. For each of such requests as ”man with braces” or ”braces smile” several hundreds of high-resolution images were downloaded. Duplicates were removed using a perceptual hashing algorithm and all images were processed and cleaned using the pipeline described above. About 1.5 thousand photos of people with braces were obtained using the described process.

Face and mouth crops were produced for each image using the image processing pipeline. Face crop is used for training a baseline model and for human evaluation of the proposed pipeline. Examples are presented on figure 3.1. Mouth crop is used to train the generative network in the pipeline. Examples of mouth crop images are shown in figure 3.2.

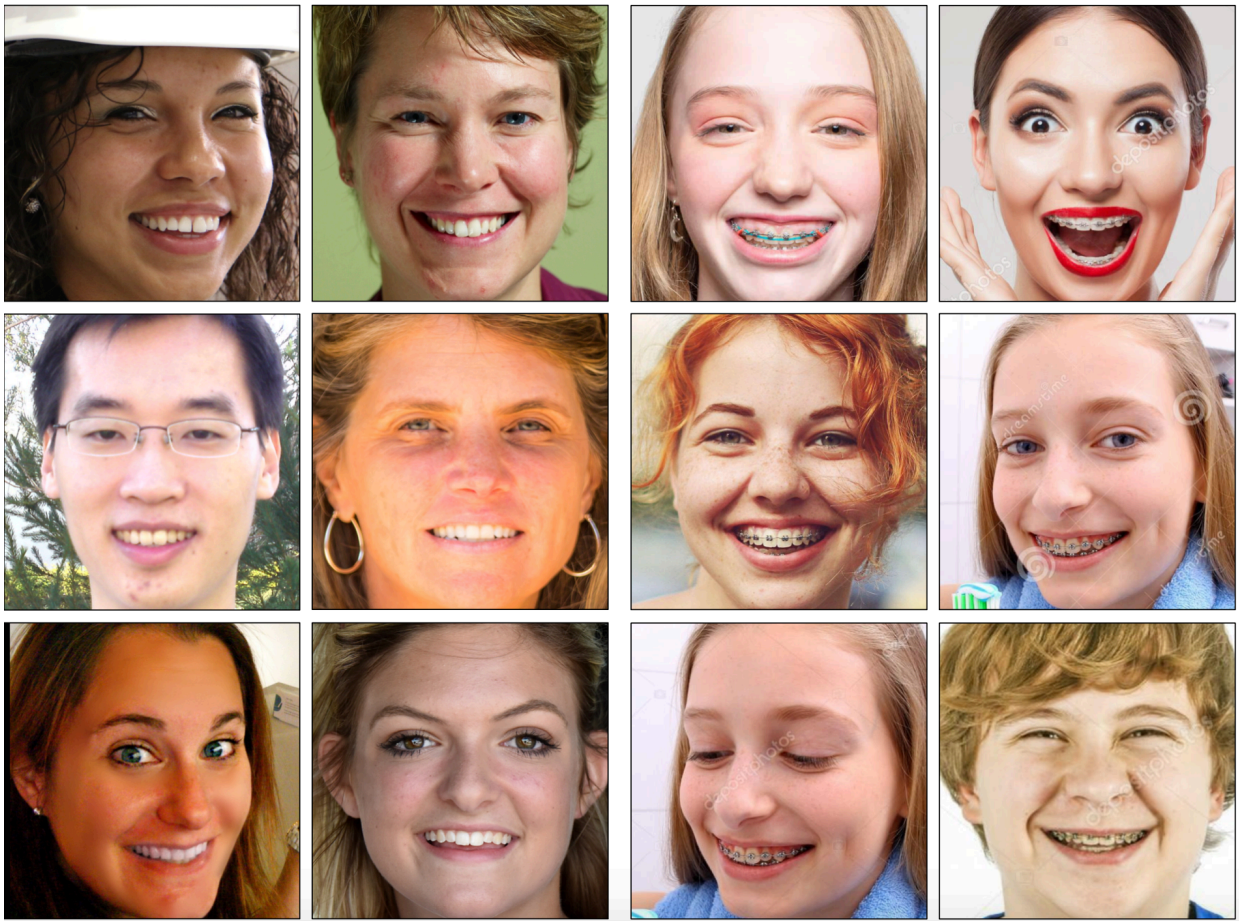

Figure 3.1: Examples of face crop images.

Figure 3.2: Examples of mouth crop images.

Chapter 4

Evaluation

4.1 Baseline algorithm

To study the performance of the proposed image processing pipeline it is needed to be compared with a simpler algorithm which results would be considered as a baseline. A reduced version of the proposed pipeline without mouth aligning process was explored for this purpose. In this algorithm a generative model accepts an image of a face with braces as an input and it should return an image of the face without braces. The dataset obtained in this work was used to train AGGAN model. Input images were resized to 256x256 and other training parameters were identical with the full pipeline. Examples of reduces pipeline results are presented in figure 4.1. Even though the model was trained to work with low-resolution examples and has a built-in attention mechanism in many cases it fails to learn face semantic and produces many artifacts on the resulting images. The reduced pipeline shows a very poor quality of the results and it is pointless to perform a direct side-by-side comparison with the full image processing pipeline.

Figure 4.1: Examples of baseline models results. Original images are presented in the first row, baseline pipeline results are in the second row, full pipeline results are in the third row.

4.2 Crowdsourcing evaluation

Quantitative evaluation of results of image-editing algorithms or image generators is complicated. Many metrics are described and compared in literature [2, 23, 19]. But still, there is no complete and comprehensive metric that would fully correlate with human evaluation. For the braces removing task considered in this work obtaining a photo-realistic result is crucial so human evaluation is needed.

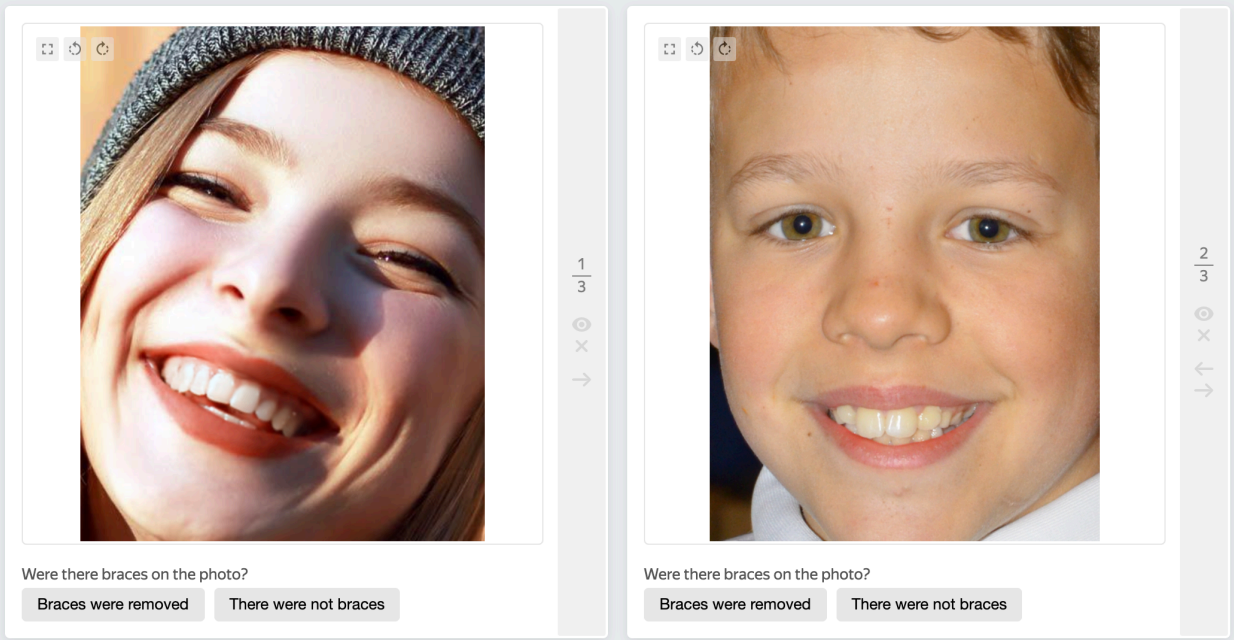

Figure 4.2: Example of Toloka crowdsourcing platform interface.

The evaluation was performed using Toloka crowdsourcing platform [7]. About 100 images of people with braces from the validation set were processed using the braces removing pipeline and then they were mixed with images of people without braces. To make evaluation more precise and to remove a distraction only faces were left on the photos. Before showing the images it was explained that some of them originally had dental braces that were removed automatically and some did not and several examples were shown. For each image there were two variants ”Braces were removed” or ”There were no braces”. An example of this interface is presented in figure 4.2 Each image was then shown to ten people and their average answer were interpreted as a human quality score in the range from 0 to 1 where ”1” means that all answers are ”There were no braces” and ”0” means that everybody selected ”Braces were removed”. Distributions of the answers for photos without braces and with removed braces are presented on image 4.3. It can be seen that distributions are not identical as they would be if the algorithm is perfect, but in 54% of cases, people fail to determine that braces were on the original image. And in 20% of cases images that were not edited were considered as edited. Confusion matrix for all answers is presented in table 4.2. Examples of validation images with braces and their human scores are presented in Appendix A.

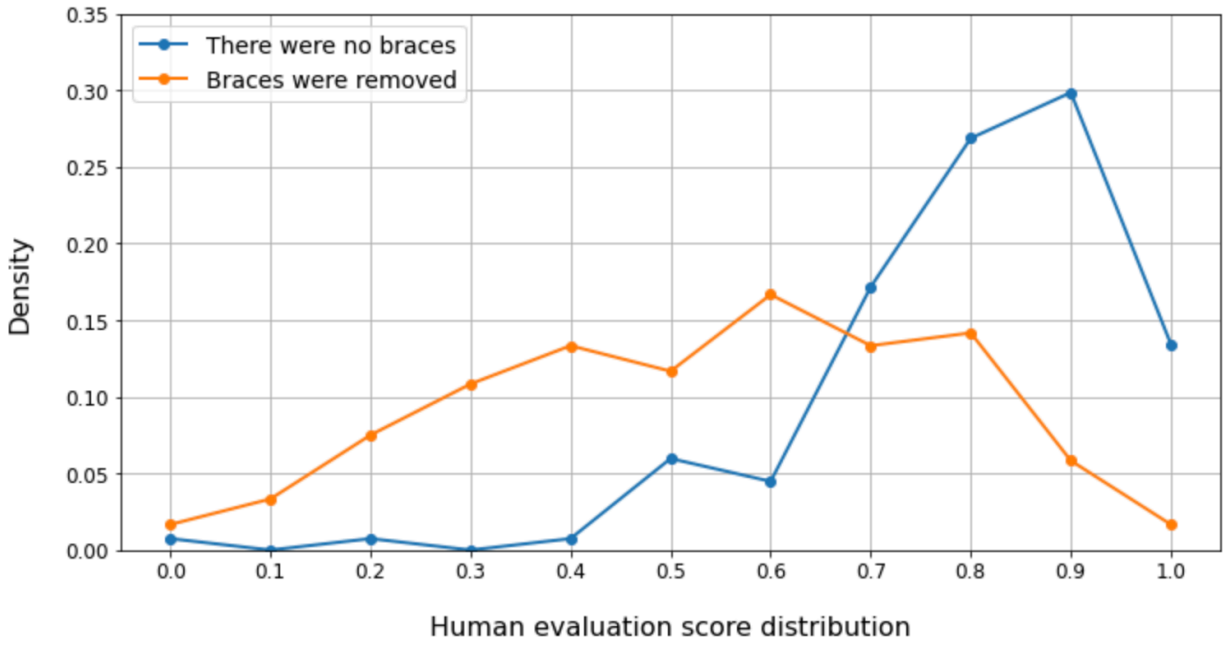

Figure 4.3: Human scores distribution for images with removed braces and images of people without braces.

Table 4.1: Confusion matrix for human evaluation of braces removing.

Answer / Label	There were no braces	Braces were removed
There were no braces	1071	269
Braces were removed	646	554

An interesting feature of architectures for unpaired image-to-image translation such as CycleGAN or AGGAN is that they have two generators for cross-domain transition. This means that during training the model for removing braces a model for generating braces was also obtained. This model can be applied and evaluated almost in the same way as the braces removing model. For this evaluation images with generated braces are shuffled with photos that originally had braces and people should select one variant from ”Braces were drawn” and ”Braces were initially”. In this human evaluation score ”1” means that everyone considered braces as real and ”0” means that braces were considered as generated. In 58% of cases, people determined that braces were generated and 31% of cases real braces were considered as generated. The distribution of answers are presented in figure 4.4. Confusion matrix is presented in table 4.2. Examples of generated results are presented in Appendix A.

Figure 4.4: Human scores distribution for images of people with generated braces and with real braces.

Table 4.2: Confusion matrix for human evaluation of braces generation.

Answer / Label	Braces were initially	Braces were drawn
Braces were initially	824	366
Braces were drawn	693	517

Chapter 5

Conclusion

In this thesis, a comprehensive multistage image editing pipeline for removing braces from a photo was developed. A new large dataset was collected specifically for the problem. The quality of the obtained pipeline was studied through a direct comparison with a baseline model and using a Toloka crowdsourcing platform. This study proved that the developed pipeline performs much better than a baseline generative model and shows sufficient result quality using human evaluation. The proposed pipeline and data preparation algorithm can be easily adapted for other face attribute editing tasks.

The pipeline is implemented as a web service in Docker and can be easily deployed on any machine. Also, the pipeline can be conveniently used through a Telegram bot. This bot is planned to be promoted and launched publicly.

Further development of the pipeline may include more experiments with models and training parameters, obtaining more data, increasing overall inference speed through optimizing pipeline algorithms.

Bibliography

[1] Grigory Antipov, Moez Baccouche, and Jean-Luc Dugelay. Face aging with conditional generative adversarial networks. In *2017 IEEE international conference on image processing (ICIP)*, pages 2089–2093. IEEE, 2017.

[2] Ali Borji. Pros and cons of gan evaluation measures, 2018.

[3] Vien Cheung, Stephen Westland, David Connah, and Caterina Ripamonti. A comparative study of the characterisation of color cameras by means of neural networks and polynomial transforms. *Coloration Technology*, 120:19 – 25, 01 2004.

[4] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[5] François Chollet. Xception: Deep learning with depthwise separable convolutions, 2016.

[6] Jiankang Deng, Anastasios Roussos, Grigorios Chrysos, Evangelos Ververas, Irene Kotsia, Jie Shen, and Stefanos Zafeiriou. The menpo benchmark for multi-pose 2d and 3d facial landmark localisation and tracking. *International Journal of Computer Vision*, pages 1–26, 2018.

[7] Alexey Drutsa, Viktoriya Farafonova, Valentina Fedorova, Olga Megorskaya, Evfrosiniya Zerminova, and Olga Zhilinskaya. Practice of efficient data collection via crowdsourcing at large-scale, 2019.

[8] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014.

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*, 2015.

[10] Zhenliang He, Wangmeng Zuo, Meina Kan, Shiguang Shan, and Xilin Chen. Attgan: Facial attribute editing by only changing what you want, 2017.

[11] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks, 2016.

[12] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution, 2016.

[13] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks, 2018.

[14] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2014.

[15] Cheng-Han Lee, Ziwei Liu, Lingyun Wu, and Ping Luo. Maskgan: Towards diverse and interactive facial image manipulation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[16] Guilin Liu, Fitsum A. Reda, Kevin J. Shih, Ting-Chun Wang, Andrew Tao, and Bryan Catanzaro. Image inpainting for irregular holes using partial convolutions, 2018.

[17] Kamyar Nazeri, Eric Ng, Tony Joseph, Faisal Z. Qureshi, and Mehran Ebrahimi. Edgeconnect: Generative image inpainting with adversarial edge learning, 2019.

[18] Jie Shen, Stefanos Zafeiriou, Grigorios Chrysos, Jean Kossaifi, Georgios Tzimiropoulos, and Maja Pantic. The first facial landmark tracking in-the-wild challenge: Benchmark and results. 12 2015.

[19] Konstantin Shmelkov, Cordelia Schmid, and Karteek Alahari. How good is my gan?, 2018.

[20] Hao Tang, Dan Xu, Nicu Sebe, and Yan Yan. Attention-guided generative adversarial networks for unsupervised image-to-image translation. 2019.

[21] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Deep image prior. 2017.

[22] Zongwei Wang, Xu Tang, Weixin Luo, and Shenghua Gao. Face aging with identity-preserved conditional generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7939–7947, 2018.

[23] Qiantong Xu, Gao Huang, Yang Yuan, Chuan Guo, Yu Sun, Felix Wu, and Kilian Weinberger. An empirical study on evaluation metrics of generative adversarial networks, 2018.

[24] Zili Yi, Hao Zhang, Ping Tan, and Minglun Gong. Dualgan: Unsupervised dual learning for image-to-image translation, 2017.

[25] Changqian Yu, Jingbo Wang, Chao Peng, Changxin Gao, Gang Yu, and Nong Sang. Bisenet: Bilateral segmentation network for real-time semantic segmentation, 2018.

[26] Stefanos Zafeiriou, Grigorios Chrysos, Anastasios Roussos, Evangelos Ververas, Jiankang Deng, and George Trigeorgis. The 3d menpo facial landmark tracking challenge. In *International Conference on Computer Vision (ICCV) Workshops*, 2017.

[27] Stefanos Zafeiriou, George Trigeorgis, Grigorios Chrysos, Jiankang Deng, and Jie Shen. The menpo facial landmark localisation challenge: A step towards the solution. In *Computer Vision and Pattern Recognition (CVPR) Workshops*, 2017.

[28] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, and Yu Qiao. Joint face detection and alignment using multi-task cascaded convolutional networks. 2016.

[29] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks, 2017.

Appendix A

Examples of the results

Human score is described in section 4.2

A.1 Braces removing

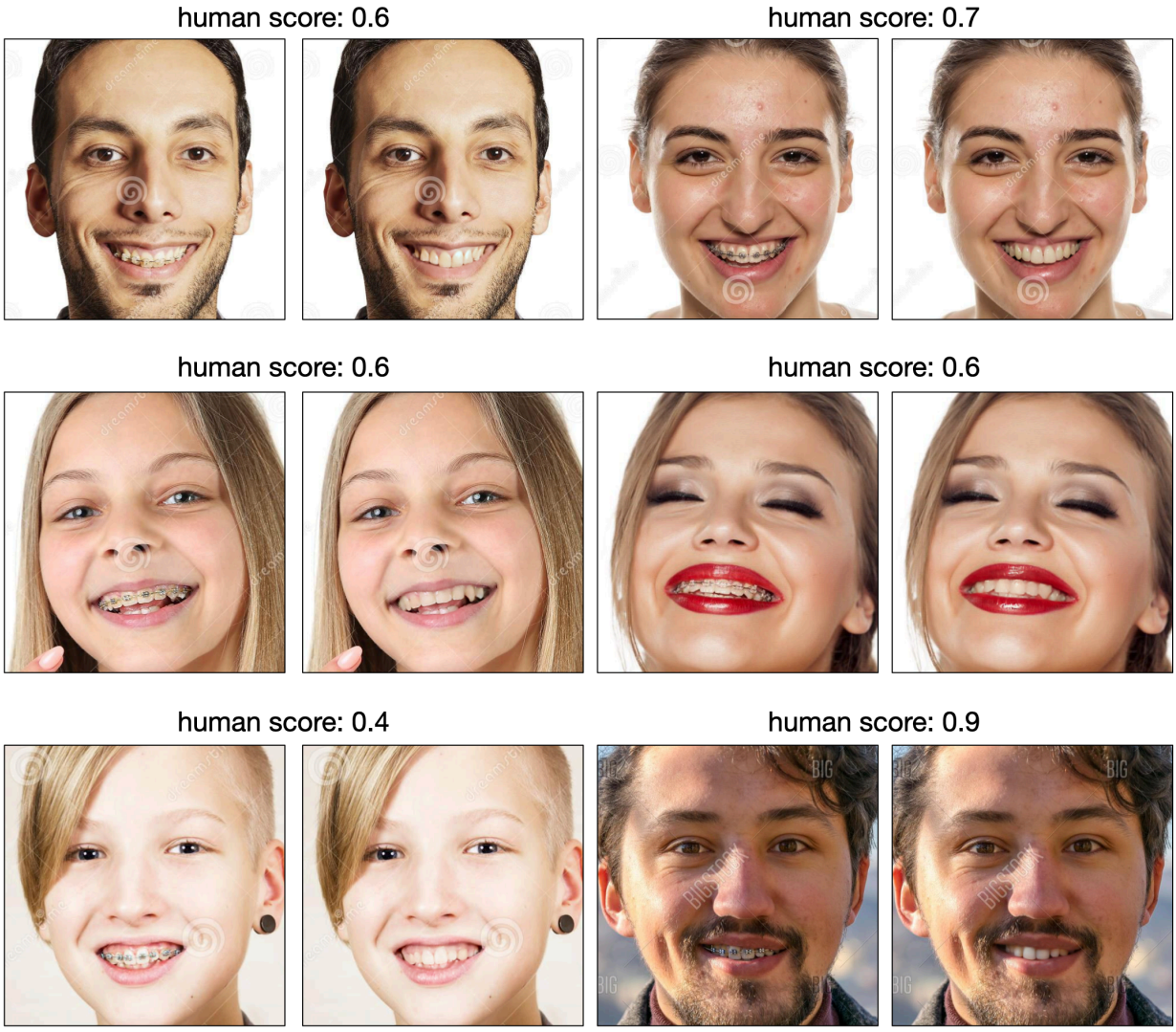

Figure A.1: Images before and after braces removing with human score for each pair.

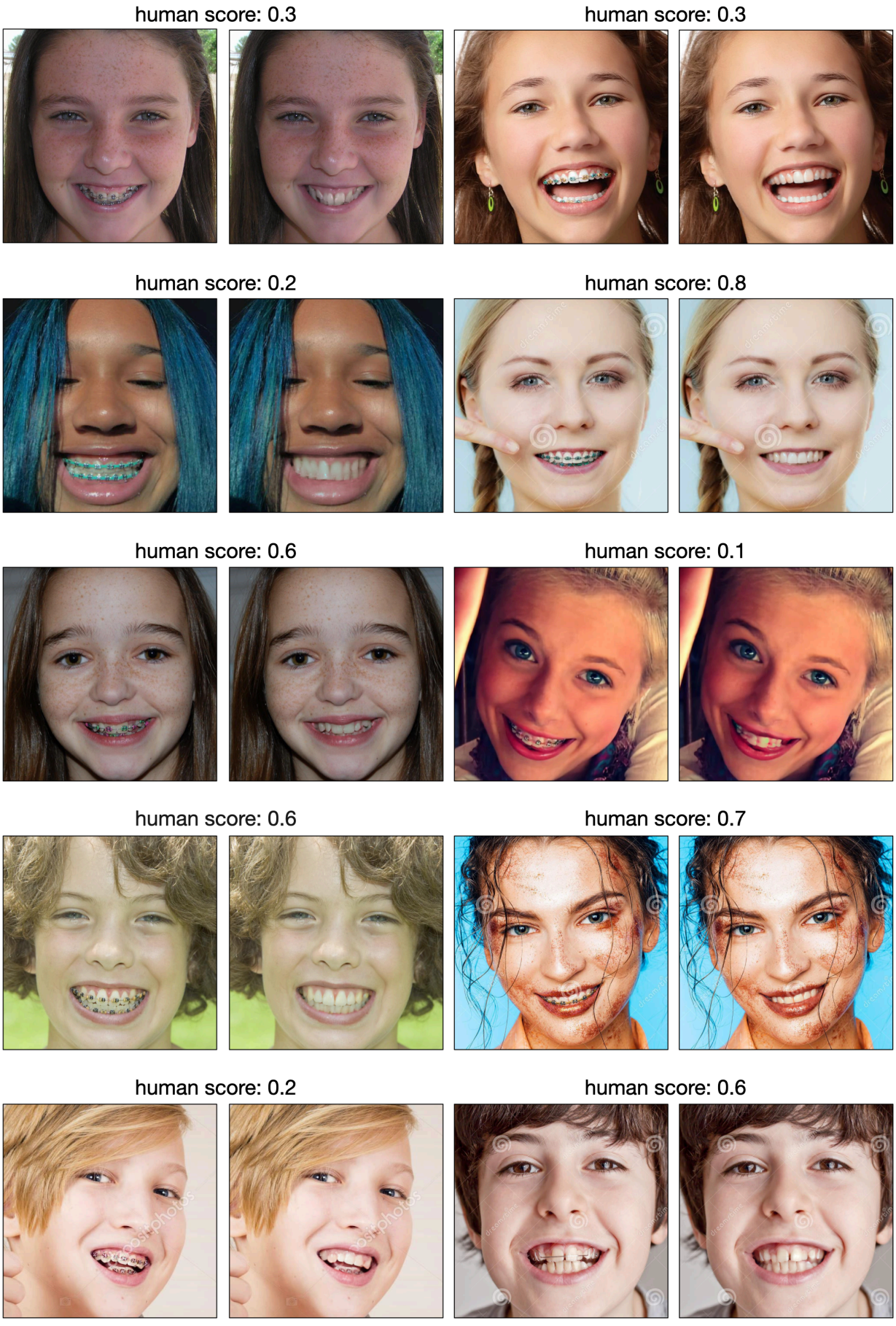

Figure A.2: Images before and after braces removing with human score for each pair.

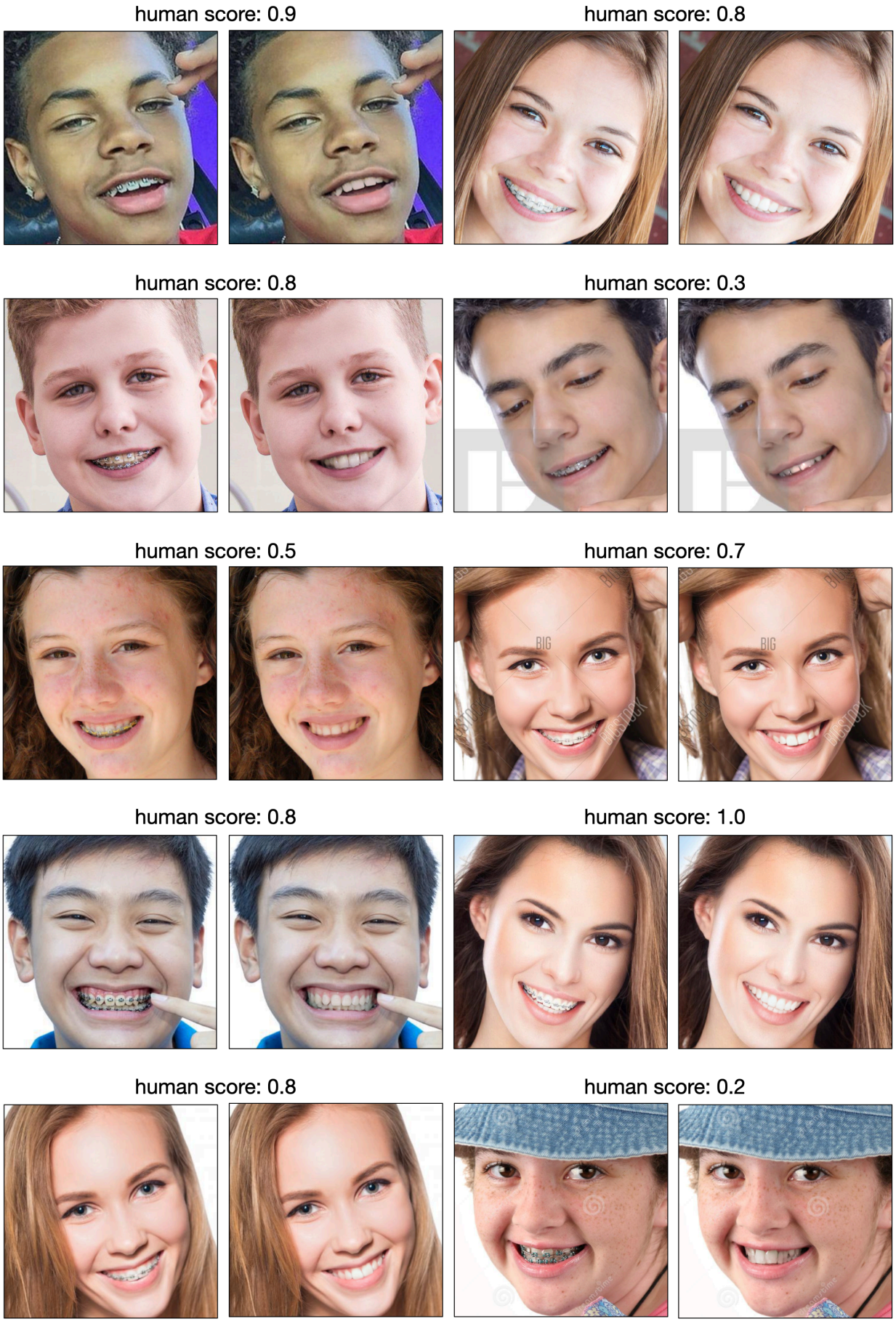

Figure A.3: Images before and after braces removing with human score for each pair.

A.2 Braces generation

Figure A.4: Images before and after braces generation with human score for each pair.

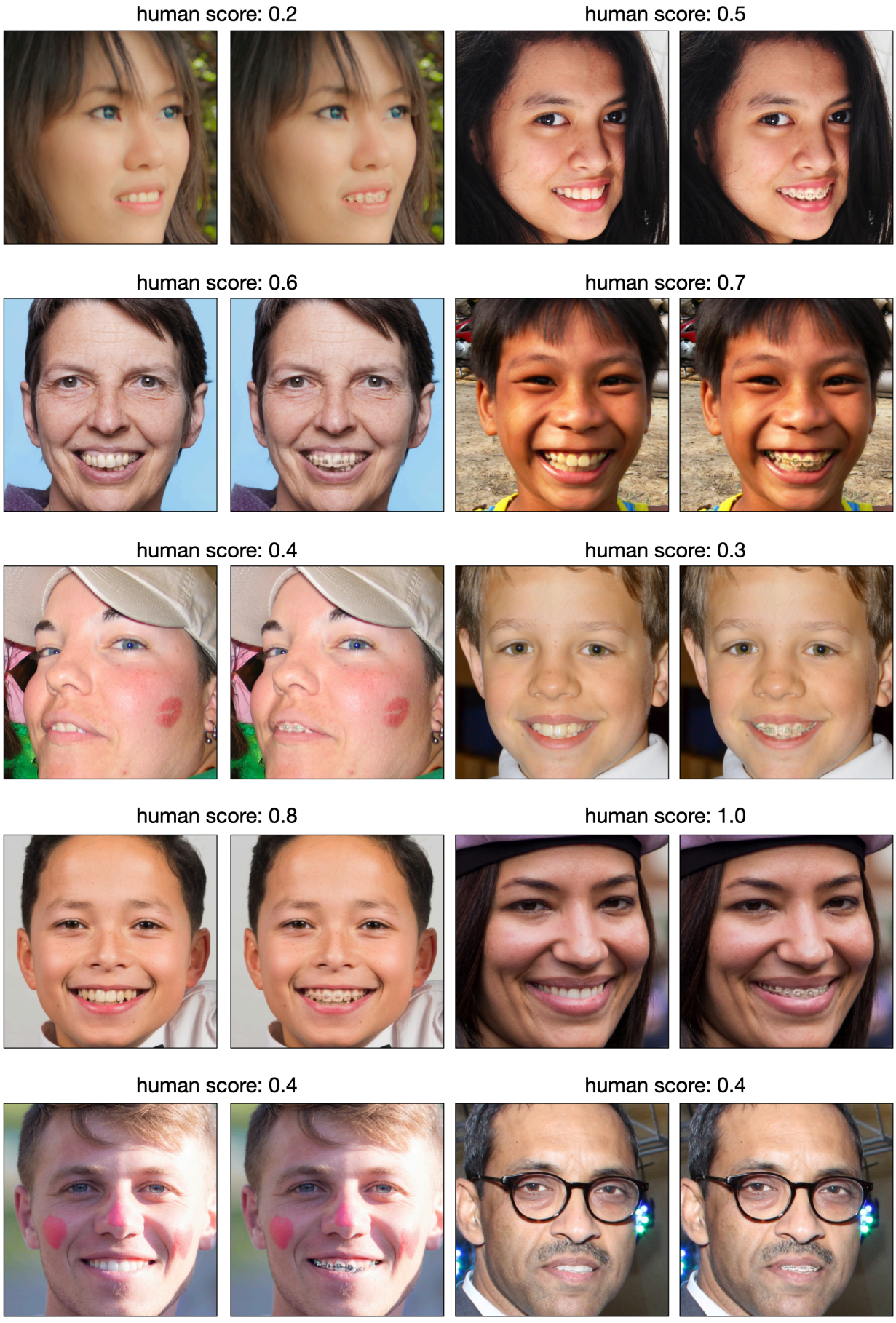

Figure A.5: Images before and after braces generation with human score for each pair.