

CE301 Individual Capstone Project Challenge - Final Report

***Developing Machine Learning Methods to
Automatically Geo-Localise Storybooks Artwork***

**Student: Anastasia-Valentina Tomita
Student Registration Number: 1801722**

**Supervisor: Dr. Shoaib Jameel
Second Assessor: Dr. Yunfei Long**

School of Computer Science and Electronic Engineering
University of Essex

April, 2022

TABLE OF CONTENTS

1	INTRODUCTION	3
1.1	Discussion	3
1.2	List of Abbreviations	4
2	BACKGROUND INFORMATION	6
2.1	Discussion	6
2.2	Cartooned Datasets	7
2.3	Element Cartoon-isation	8
3	METHODOLOGY	12
3.1	Object Segmentation	12
3.2	Contextual Object Similarity	15
	3.2.1 Emotion Detection	18
	3.2.2 Gender and Age Analysis	22
	3.2.3 Body Position	24
3.3	Matting and Compositing	24
4	PROJECT PLANNING	28
4.1	Project Research and Planning	28
4.2	Jira	28
4.3	GitLab	30
5	RESULTS AND ANALYSIS	32
5.1	Analysis	32
6	CONCLUSION	34
6.1	Further Research	34
6.2	Retrospection	34
6.3	Conclusion	36
	REFERENCES	36

Acknowledgements

I would like to thank the University of Essex and the School of Computer Science and Electronic Engineering for its support in my development and also for providing me the necessary equipment and tools required for this project to be possible.

I also want to thank Dr. Shoaib Jameel for his support during my project. He provided me with the necessary advice and resources required for me to achieve my most important educational accomplishment to date.

Abstract

Modifying a certain book in order for it to be accessible to different parts of the world is nothing new. However, when it comes to illustrations this process gets more complicated, and it is often overlooked due to its complexity. For young readers, the illustrated books are the stepping stone in their language development, whilst also expanding their knowledge about the world around them. But because the world is very culturally diverse, it is often the case there is an inconsistency between how children perceive their world and the fictional world the book creates for them. For this reason, the text understanding is often much more difficult, children not being able to relate to the book content. This issue brings the motivation for my final year project, that tries to bring a more natural experience to storybook reading, by presenting the first steps towards a geo-localised image translation algorithm.

Index Terms - Geo-Localisation, Cartoons, Facial Emotion, Style Transfer, Matting, Compositing, Deep Learning

INTRODUCTION

1.1 Discussion

Image geo-localised translation is not a common topic within the Computer Vision community. Whilst there are proposed works [1] [2] able to extract and translate text from images, there are yet to be seen cases of visual geo-localisation. In order to modify a certain cartoon, one needs the skills of a visual artist, the frame being modified either manually as seen in *Figure 1.1*, either through the use of graphics editors such as Adobe Photoshop or Lightroom. Such a process is often very time consuming, reason for which this project looks into the possibility of autonomously geo-localising frames within the context of children's illustrations.

In order for the process of automatic image translation to be possible, several steps have to be completed, such as: object detection and localisation, instance segmentation, matting, compositing, emotion detection. Recent papers exemplify the state-of-the-art achievements on all the above mentioned topics. However, all these are presented as separate research topics and not within a larger framework able to automatically or partially automatically modify a given frame in order for it to fit other cultural backgrounds. Therefore, this project proposes a novel framework that is able to find the similarity between human instances, taking as key reference the output from the emotion detector and the age and gender detection module. Furthermore, such frameworks are often used on frames depicting the real world, where there is a significant level of uniformity. On the other hand, cartoons are known to present a high level of abstraction, due to them being purely fictional and mainly designed for children's entertainment. Geo-localised image translation is significantly challenging due to the variations in the above-mentioned abstraction level. The way in which the characters, events and emotions are being presented is highly dependant on the vision of the visual artist and on the distinctive signature that they present throughout their art. Despite the fact that these patterns are often easily observed by most human readers, they pose significant problems for autonomous systems in the absence of extensive training data that presents all the distinct drawing styles.

For the purpose of this project, the emphasis was put on the human instances alone as they require a more complex translation process, being the principal emotion transmitters within any given frame. However,

what can we categorise as being a human instance? Due to the fact that cartoons are purely fictional it is often the case that inanimate objects become main characters in order to teach the young readers about care and empathy to the world around them. In order to establish emotional connections between the reader and the characters, the objects are brought to life through the power of motion and through being empowered with free thought and emotions. Because of these human-like traits, in order for the objects to become more realistic, they often appear to be adjusted to fit within our concepts of a living individual: having facial features and expressions, arms and legs, and other elements which give them a human aspect. This anthropomorphisation of inanimate objects, alongside with the appearance of transformed biological lifeforms such as talking donkeys introduce confusion for any autonomous system when analysing a certain frame. This is due to the high similarity between the representation of human and object instances within illustrations. However, having characteristics of the humankind do not categorise the above as being human.

The remainder of the project report is organised as follows: Section 1 presents an overview of the background information related to the project's topic. Section 2 presents the component parts of the geo-localisation algorithm, while in Section 3 the project planning is being discussed, alongside with the management tools used. Section 4 contains the results and analysis, whilst the conclusion and future work are presented in Section 5.

1.2 List of Abbreviations

- CNN - Convolutional Neural Network
- DNN - Deep Neural Network
- NST - Neural Style Transfer
- HI - Human Instance
- AI - Artificial Intelligence
- SD - Standard Deviation
- ED - Euclidean Distance

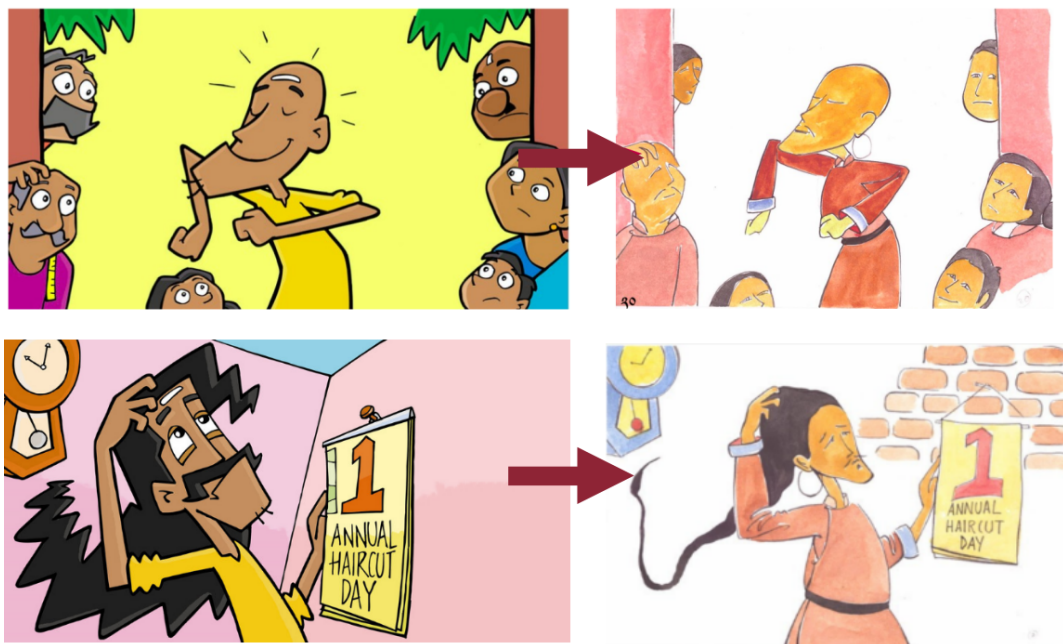


Figure 1.1 - Example of manual translations of a given frame. Images taken from [3].

BACKGROUND INFORMATION

2.1 Discussion

Automatic image translation includes several well-defined sequential steps, namely image segmentation, gender and emotion detection, matting and compositing, which are being presented in more depth within *Chapter 3*. A combination of hand-crafted approaches and deep learning techniques are being fused together in order to extract the image content, process its meaning and find an adequate replacement for each element. Hand-crafted or engineered features refer to the properties of an image which are being extracted through the use of certain manually predefined algorithms [4]. Such frameworks analyse the various image patterns which can be detected through observing abrupt changes within adjacent pixels. For instance, two simple features that can be extracted through the use of such algorithms are edges and corners, which are further used to extract the replacing objects from their background, process known as matting.

A task such as image translation cannot be applied to the image as a whole, but rather to its individual component parts. In order to preserve the meaning of the frame, we have to understand the content of separate areas of an image with the purpose of replacing them with their corresponding best match, without modifying overall object placement. In such a case, more complex features (such as textures or component parts) need to be extracted in order to recognise certain areas of the image as representing a particular object. Convolutional Neural Networks (CNNs) are able to detect such complex features, which are being learnt during the training process of the model by observing similarities between multiple objects of the same type. Image segmentation uses such a neural network and is one of the key methodologies used within this project, as it provides a pixel-level delimitation of objects whilst also detecting their respective meaning. By further analysing the detected area, we can extract additional meaning from the image content. When it comes to illustrations containing human instances, it is crucial to maintain certain physical characteristics such as age, gender and transmitted facial emotions in order to maintain the overall meaning of the frame without interfering with the narrative plot. These extracted physical features are being compared with the ones resulted by analysing the stored dataset images in order to find the best match for replacement.

2.2 Cartooned Datasets

The fact that illustrations have yet to be completely understood by autonomous systems is mainly due to the limited amount of cartoon-related data which has yet been put together. Any deep learning technique needs large amounts of data during its training process in order to observe patterns within multiple objects of the same type, consequently being able to detect previously unseen objects based on these learnt features. As opposed to the real world which is easily predictable, where every shape can be mathematically represented and perceived, illustrations have something uncommon, and that is because they are highly dependant on the imagination, skills and vision of the visual artist. Because of the signature of the creator, a certain image can be cartooned in different ways and a certain emotion can be represented in a completely distinct manner. So far, we have observed that computers have achieved incredible results when it comes to them perceiving and understanding the real world, and this is due to its predictability. Because of this fact, most cartoon datasets tend to mimic this natural regularity by only analysing matching styles and do not explore multiple styles within the same context. Such a pattern can be observed in the work of Jain Nikita *et al.* in [5], where facial emotions within cartoon frames have been analysed, and where a custom dataset has been created, depicting different scenes from the well-known "Tom and Jerry" cartoon. Their proposed Deep Neural Network (DNN) approach achieves great results when it comes to the 4 defined emotions, namely happy, sad, angry and surprised. This is due to the above-mentioned reasons - all these emotions follow a predictable pattern which is easily learnable through the use of DNNs.

Not many cartoon datasets have yet been created, and as mentioned above, they are too small and too precise in order to fit different types of styles. Also, most are not openly available to the public, such as the dataset published by Jain Nikita *et al.* in [5], thus not being able to build upon. As opposed to the above, the only publicly published dataset is iCartoonFace, which was released in 2021 by Yi Zheng *et al.* in [6] and which comprises a wide variety of cartooned characters, within different artistic styles. As it is designed for the task of facial recognition within illustrations, it lacks instances of every-day cartooned objects, so it would be insufficient when it comes to whole-image geo-localisation. For the purpose of this project, as the emphasis falls onto human instances replacement, I have created a sub-selection from [5], which contains 160 facial expressions within cartoon frames. Because the iCartoonFace dataset was only created for facial recognition and does not contain the emotions presented by each character, manual annotation has been undergone for the above-mentioned images, presenting the ground-truth for the facial emotion within each frame. I have used the above mentioned dataset to test the emotion, age and gender modules, results of which are being presented in *Section 5*. Frames taken from the dataset can be found in the Appendix.

Furthermore, when translating a frame from one region to another, one would need a dataset of cartooned

characters from different countries around the world, which would accurately depict the traditional clothing, objects and human features which represent the people from each region. However, such dataset is not currently available. Therefore, I have manually collected as many cartooned references as openly available, that have the above mentioned traits, which can be found in the appendix. Whilst the cartooned characters depict region-specific traits, they do not present a wide variety of body positions and emotions due to their informative and educative scope, being initially created to show the traditional attire from different countries around the world. Due to this fact, I have also looked into character cartoon-isation, as presented in *Section 2.3* below.

2.3 Element Cartoon-isation

To automatically geo-localise a certain illustration, one needs to keep a similar level of abstraction throughout the frame in order to bring realism to the translated replica. Also, one needs to transform each replacing object in such a manner that it would fit with the remainder of the frame, such that it would not have a distinctive appearance. Due to the lack of extensive illustration datasets which are currently available, in order to bring a more story-like appearance to a certain frame, one can modify real life objects to match the style of the initial figure. Through the use of Feed-forward Convolutional Neural Networks, Gatys *et al.* in [7] propose a model able to adapt a picture’s style, whilst keeping the content unchanged. The proposed work demonstrates that a separate manipulation of the content and style representations of the image is possible, due to the structural arrangement of the content within the layers of the CNN. The more we progress through the CNN’s architecture, the content information is increasingly explicit, observing a shift in attention from low-level features such as blobs of colour to more complex features containing more and more contextual information. Neural Style Transfer takes advantage of this hierarchical feature structure within CNNs, minimising the differences between the low-level patterns of the inputted image and the style extracted from the reference image [8]. Therefore, such approaches [7] [9] [10] are able to modify the patterns within a certain frame, whilst maintaining its context unchanged. The architecture proposed by Gatys *et al.* in [7] is presented in *Figure 2.1*, where the shift in image style can also be observed.

By using [11], I have experimented with different styles in order to determine which ones produce a better cartoon-isation. I have observed that, during the task of stylistic pattern alteration, one needs to introduce more shaping to the frame, such as rough edges and colour patterns. I have selected two styles which are presented in *Figure 2.3*, one which brings more abstract shaping to the images and another one which transforms them in such a manner that they appear drawing-like. I have also experimented on overlapping multiple styles within a single frame, although the desired output was never reached. This is

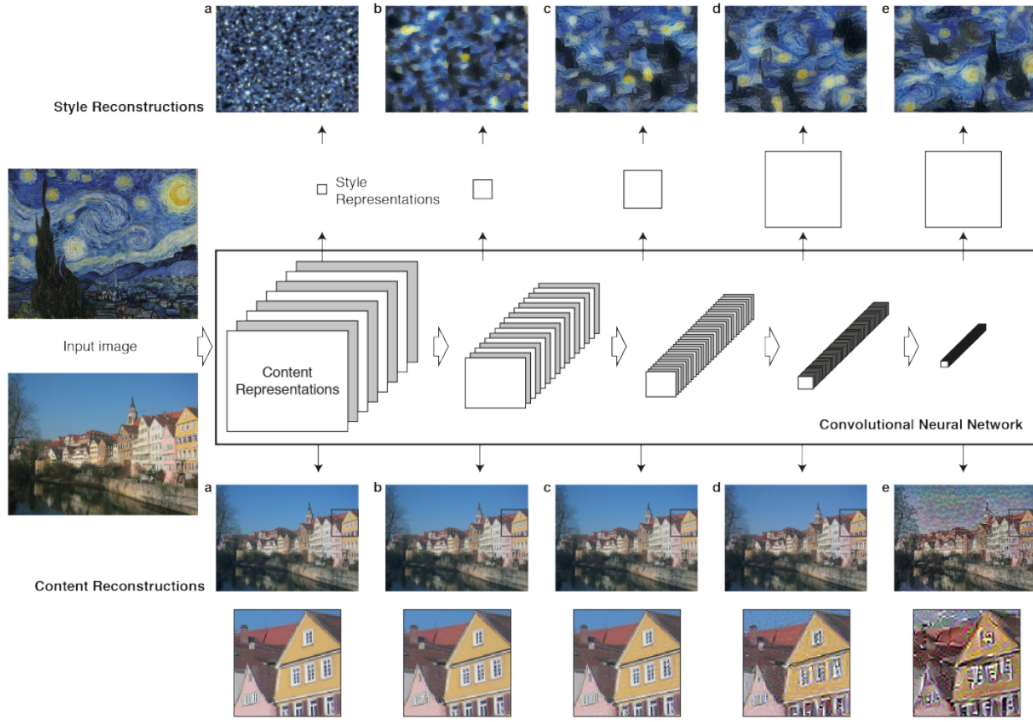


Figure 2.1 - Convolutional Neural Network architecture, used for the task of Neural Style Transfer [10].

because the final image will always resemble the last style applied, which would make the application of the previous styles redundant.

One weakness of the NST approach is the fact that the outline of the objects and individuals within the initial frame remains constant throughout the transition process, because the changes are produced only when it come to the low-level aspects of the image. Therefore, individuals remain within their natural proportions, without the abstraction often encountered within cartoons. However, there are yet no approaches that alter the shaping of objects, which would need to use the features extracted at higher levels of the CNN architecture, modifying the content whilst also ensuring that the its meaning remains unchanged.

Furthermore, Wang *et al.* in [12] presents a model for whole-image cartoonisation, that identifies textures, flattens the content of images and smooths surfaces in what they introduce as "the three white-box representations". Output from the above-mentioned model can be observed in *Figure 2.2*, where images of individuals with different traditional attires are being modified in order to have a more cartooned appearance. I have also experimented on further applying the NST technique to examine the changes, and I have observed the natural colours being gradually altered, whilst the image content is swiftly being lost, specifically within regions with higher level of texture. Moreover, I have observed that the technique presented in [12] has better output when applied on individuals that have clothing with bright colours, thus having a high contrast.

In [13], Yong Zhang *et al.* propose another approach to person cartoonisation, which extracts the facial

features of an individual, and retrieves the most similar cartooned facial component for each one of them. After doing so, all these segments are adjusted and incorporated in order to form the final replacement. However, the approach is not yet available to the public, therefore could not be tested and integrated within this project.



Figure 2.2 - Exemplification of the output extracted from the person cartoonisation technique, which is able to modify a frame such that its content would appear more story-like. However, bright colours are needed to be present in the original frame in order for the blending process of the method to not remove too much from its content [12].

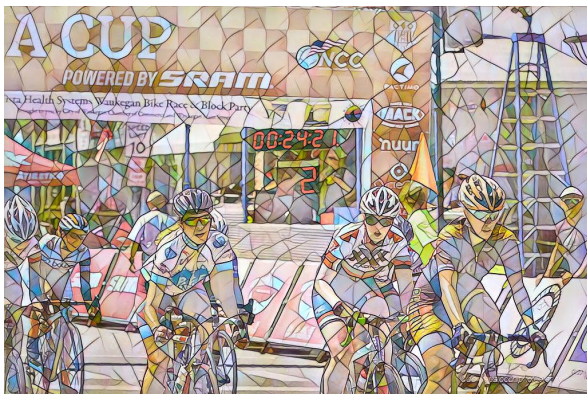
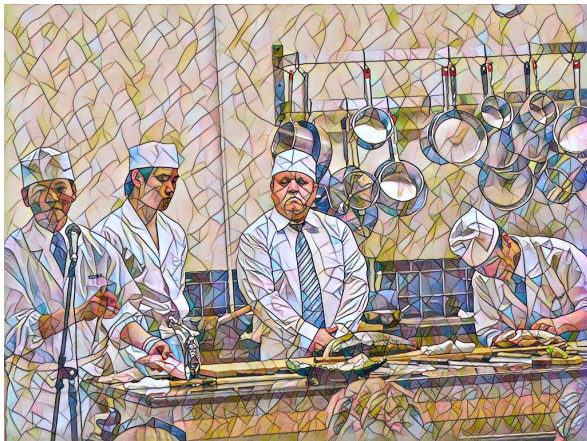
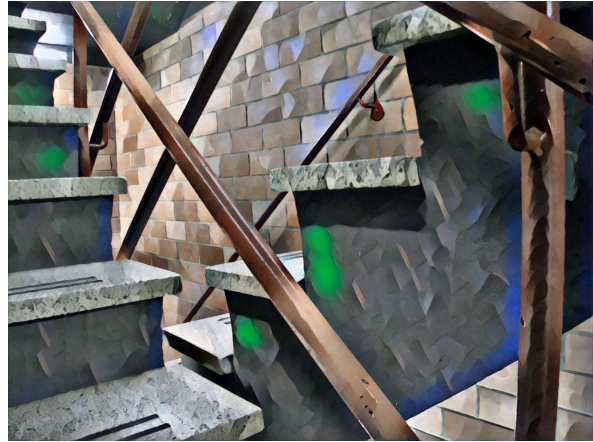


Figure 2.3 - Two image styles are being presented. On the left such a style introduces more shaping into the initial frame, making it to look like a cartoon, whilst the second style modifies the image such that it appears more story-like. The images are extracted from the Open Images Dataset [14] and altered through the use of style transfer algorithms [11].

METHODOLOGY

3.1 Object Segmentation

To autonomously modify the digital content of a story during its translation process, the distinct aspects between different locations should be taken into consideration, including any cultural and geographic variations. Every aspect of the frames should be modified per the above mentioned variations. In order to do so, each different object within the inputted frame should first be detected, categorised and delimited through the use of deep learning models. For each detected instance, a segmentation mask would be generated which would conclude in a pixel-level separation of the given object. Several frameworks have been designed for the task of object instance segmentation, which fuses together two well-researched tasks within the Computer Vision community: object detection and image segmentation [15]. Object detection identifies the category to which the objects within a frame belong to, whilst also providing information regarding their spatial location. Image segmentation builds upon the previous-mentioned task, and can be defined as the process of dividing the pixels of an image in regions, based on their contextual meaning. In this manner we can extract information, by distinguishing different types of objects within a frame. However, such pixel separation can be done in different ways, depending on how this extracted data needs to be visualised and further processed. One such way would be through semantic segmentation [16], which groups together all pixel areas based on their detected category. On the other hand, instance segmentation [17] not only detects the class of a given object, but it is also able to distinguish between such objects, presenting them as separate entities. The difference between the two concepts mentioned above are exemplified in *Figure 3.1* below.

For the purpose of geo-localised translation, instance segmentation is being used, as it is crucial to distinguish between the characters of a cartoon. Therefore, we analyse each object individually, finding their respective replacements which best resemble the initial human instance. By doing so we ensure that the chosen replacements do not become duplicate, fact which would lead to a less natural image translation. By extending the work of Ren *et al.* in [20], Mask R-CNN [19] uses a fully convolutional network in order to predict segmentation masks with a high level of accuracy, being one of the most successful mask-based

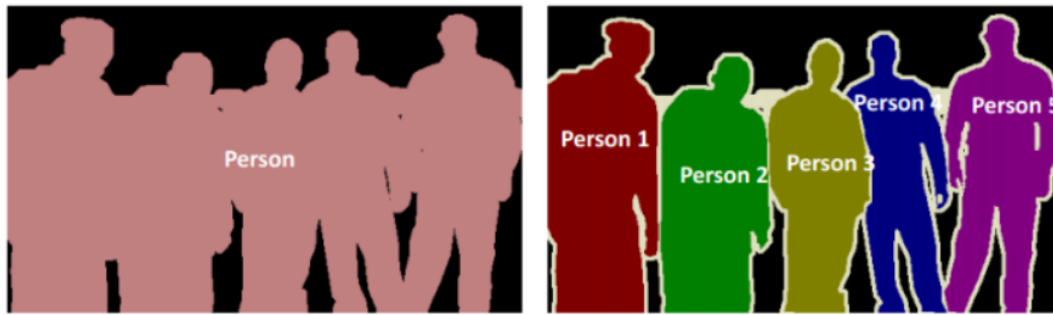


Figure 3.1 - The difference between Semantic and Instance Segmentation: On the left an example of semantic segmentation can be observed, where all objects of class "Person" which are detected are grouped together within a single instance having only one segmentation mask. On the right, instance segmentation is exemplified on the identical image. We can observe that each human instance is being detected within the corresponding class, whilst also maintaining their individuality [18].

techniques to date. CNNs [21] are currently the most effective deep learning algorithms that are used on diverse computer vision tasks to analyse visual imagery. CNNs are inspired from the neural connectivity pattern within the human brains and consist of several connected layers that can be perceived as having a 3D arrangement of neurons. Mask R-CNN uses such CNN architecture which was trained on the MS COCO dataset (Microsoft Common Objects in Context), a large-scale object detection and segmentation dataset containing 330k images, with 80 object categories and 1.5 million object instances [22]. Such an extensive segmentation data facilitates a thorough model training, ensuring a better accuracy on the provided classes. However, the downside within the project's topic is the fact that the COCO dataset presents objects as they are, from different view points and different lightning, but does not present any level of abstraction, which is often seen within children's illustrations. However, Mask R-CNN still achieves great results even on cartooned frames, as we can see in *Figure 3.4*, but there are also cases of human instances not being detected, but instead being recognised as every-day objects, as observed in *Figure 3.2*. In this case a human instance is detected as being a fire hydrant due to its unusual shape, whilst all the other detected objects being well categorised. In order to address this issue, further training is needed, such that the model would recognise the visual elements, regardless of their abstract shape. In order to do so, one would preferably have a large dataset of annotated cartooned images. In the absence of such a dataset, I have worked on styling [11] already existing frames, more precisely Google's Open Images Dataset [14] in order to bring a more cartooned appearance to the images. The biggest challenge I have encountered was that such a process is very computationally expensive, thus, when applied on a large dataset such as [14], it takes an extensive amount of time to complete. Having 1,743,004 images that needed to be stylised in two manners (as exemplified in *Section 2.3*), with 64.3 seconds per frame, the process would have taken 3.55 years to finalise, thus being completely unfeasible within this project's time frame. Therefore, I have worked on solving this time issue, through image resizing and image compression, whilst analysing the trade-off

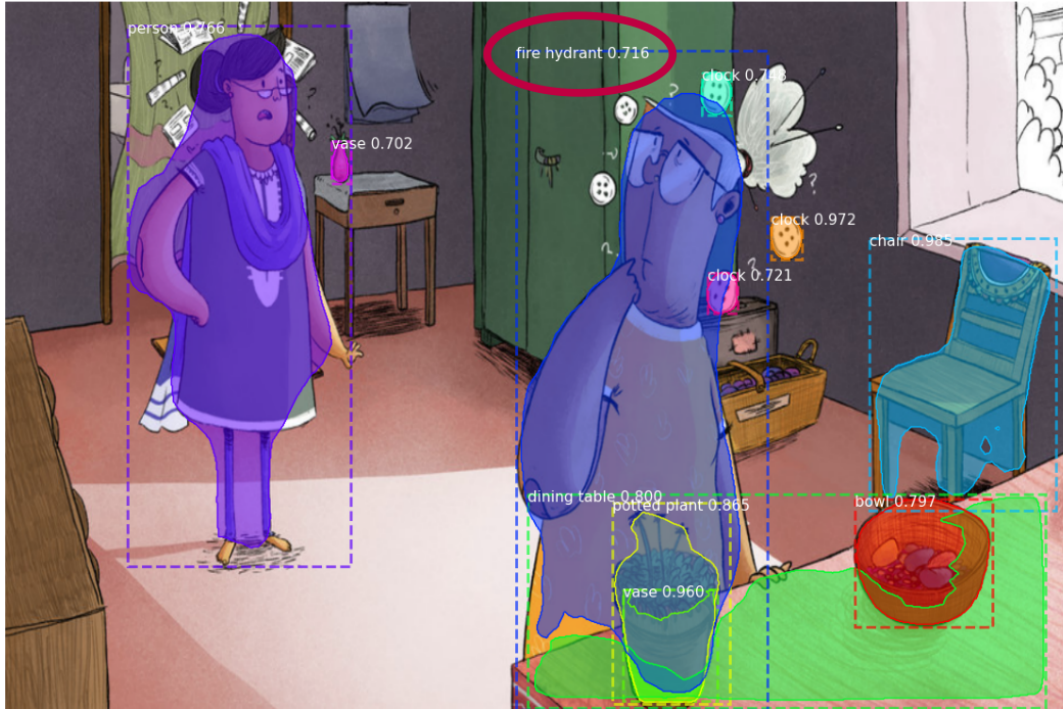


Figure 3.2 - Case in which the model wrongly detects a human instance as being a part of the "fire hydrant" class due to its unusual shape. Image taken from [3] and segmented through the use of [19].

between speed and efficiency. Image compression techniques such as JPG, JPEG and PNG [23] are able to reduce the computational cost of image processing at the cost of image clarity. Compression techniques are usually applied in such a manner that they would produce minimal quality degradation, such that it would not be perceived by users. However, images could be compressed with different levels of intensity, which I have analysed to observe the time reduction that was produced. When downgrading an image to 250x250 and a level of compression of 90%, the time needed to process it reduced to half. Moreover, after analysing the model in more depth, I have observed that the style was extracted for each frame separately, despite the fact that the dataset is being stylised with the same style throughout. Such an approach is beneficial when dealing with small datasets or when each image is being styled with different styles, but it drastically slows down the process, having a negative impact when managing large datasets. Therefore, I have applied a different method through separating the two processes - creating the module and applying it to the content images [11]. By doing so, the styling process is accelerated considerably, each image taking under a second to be modified, which has allowed me to successfully stylise the Open Images Dataset [14] as presented in *Figure 2.3*.

When it comes to image geo-localised translation, the class in which an object is being categorised is very important, as it is vital that the object is being replaced with another one which encloses unchanged context. I have provided the user with the possibility to choose what classes would be detected, as observed from *Figure 3.3*, where in the first frame we identify multiple object categories, whilst in the second frame

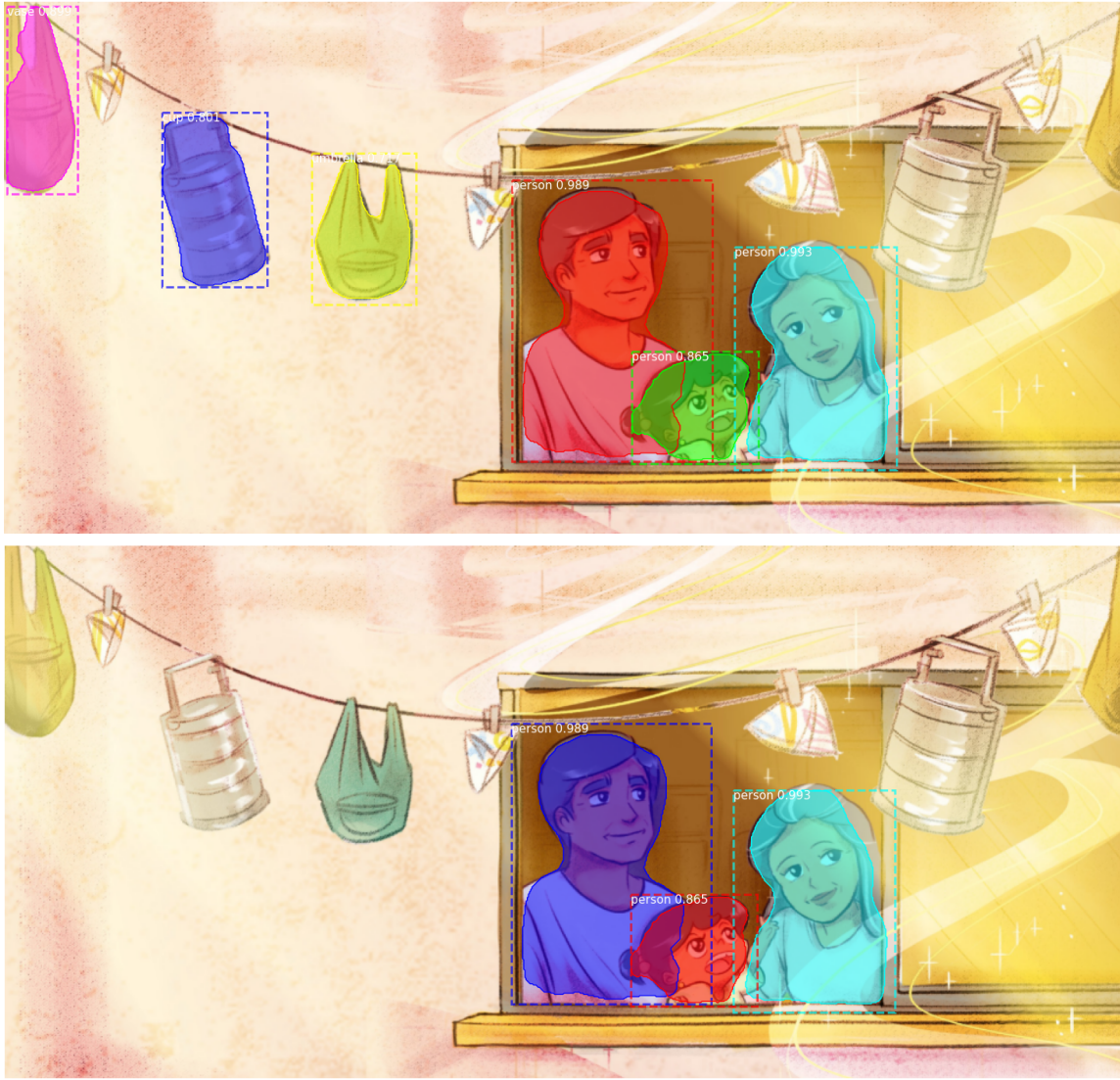


Figure 3.3 - Example of class-based image segmentation.

we only detect objects belonging to the class "person" . This is further used to only analyse the human instances within each frame, by leaving out the other detected elements. This approach can be used if a whole-image translation is desired, in order to analyse each category individually based on their priority within the image. This is because objects such as vases should always be placed on top of tables, thus table objects needing to be analysed first.

3.2 Contextual Object Similarity

For the purpose of this project, the emphasis is put on the human instances alone as they require a more complex process when replaced, due to them being the principal emotion transmitters within any given frame. Therefore, their replacement will be strongly dependant on facial emotions, gender, age and body position, which have to be detected before making the choice for a substitute. All of the above need to be



Figure 3.4 - Cases of well segmented frames. Images are taken from [3] and segmented through the use of [19].

taken into consideration when replacing the initial object, in order to keep a more natural feel to the frame and to maintain the initial context as much as possible. Therefore, in the subsections found within this chapter, detailed discussions and exemplifications can be found for each one of the three modules which collectively depict the state that characterises a human instance within a given frame.

Through the use of Euclidean distance (ED), the above-mentioned features of two distinct human instances are being compared in order to determine the contextual similarity between the two, which constitutes their matching score. This is being applied between each detected object and all the stored images, as it can be observed in *Algorithm 1* below. We always keep stored the best matching score as reference, alongside with the image path, and compare each further human instance against it.

Algorithm 1: Finding the Best Object Replacement

Given: **class_selection** set to "person"

for *object* **in range** (*number_of_detected_objects*) **do**

best_match $\leftarrow$ Null

if *class_name*[*object*] = *class_selection* **then**

 (emotion, gender, age) = detect_emotion(*object*)

for *replacement* **in range** (*number_of_replacement_images*) **do**

 emotion_r, gender_r, age_r = detect_emotion(*replacement*)

matching_score = compute_ED(emotion, emotion_r, gender, gender_r, age, age_r)

if *best_match* = Null **or** *matching_score* < *best_match* **then**

best_match $\leftarrow$ *matching_score*

best_match = **max** (*score_list*)

Within *Algorithm 1*, Euclidean distances in a higher dimensional space are being used to measure the amount of variation between two sets of values, thus the "distance" between two human objects extracted from different frames [24]. This distance shows how far apart the two objects lay on the scale of similarity, a small ED indicating that the two would be suitable for replacement, whilst a large ED would indicate the opposite. Hence, we use the detected age and gender for each pair of human instances, alongside with the certainty of its detected facial emotions by using the formula 3.1 presented below. P_{e_1} and P_{e_2} represent the probability of the emotion e to have been detected for the two objects that are being compared (labelled 1 and 2), where e ranges from 1 to 7, representing the 7 universal emotions detected, as further discussed within *Subsection 3.2.1*. A_1 and A_2 represent the age detected for the two human instances, whilst G_1 and G_2 represent their gender.

$$ED = \sqrt{\sum_{e=1}^7 (P_{e_1} - P_{e_2})^2 + (A_1 - A_2)^2 + (G_1 - G_2)^2} \quad (3.1)$$

3.2.1 Emotion Detection

When it comes to facial emotions detection, most such approaches focus on the so-called seven universal emotions: neutral, happiness, sadness, anger, disgust, fear and surprise [25] and achieve high accuracy on most public datasets [26]. However, as previously described in *Section 2.2*, there are no openly-available emotion datasets based on facial expressions within the topic of children’s illustrations. In order to train a CNN for our translation tasks, one needs a large-scale dataset of labelled images, which depict different drawing styles and different abstraction levels. Within children’s illustrations, emotions are often ambiguous and very dependant on the given context, as it can be seen in *Figure 3.5*. Thus, taking a frame out of the narrative thread without having known the background information can have a baffling effect for any autonomous system. Moreover, it is often the case that a detected emotion is not easily categorised through the use of universal emotions, which are very applicable to humans but are often not broad enough to integrate all emotional states of cartoon characters. This fact can best be observed in *Figure 3.6*, where the emotion transmitted appears too complex to be categorised within a certain state, having taken out the environmental knowledge provided by the sequential narrative aspect of cartoons. In order to address this challenge, one should not only analyse the predominant transmitted emotion, but also the possibility of each one of the seven universal emotions mentioned above. For this reason, the human instance comparison module described in *Algorithm 1* takes as input all the above-described emotion possibilities in order to conduct a better analysis of replacement fitness.

Because cartoons are intended to teach children about the world around them, the transmitted emotions are often being exaggerated in order to strengthen the young viewers’ emotional awareness and to build emotional connections, thus empathising with the characters. This exaggeration also helps to finely delimit each distinct emotional state, which in the real world are otherwise presented in a more subtle manner, harder to grasp. This awareness and understanding of human emotions would be the first step towards building emotional intelligence, which is vital for any individual in order for them to integrate into their society. Also, this exaggeration helps in making the content more enjoyable, through the amplification of emotions, states, reactions and an overall intensification of events, as observed in *Figure 3.7*. This keeps the reader entertained whilst also making the scenes more memorable. Due to all of the above, it is often the case that, when having extreme emotions, characters would change completely in shape, their facial features



Figure 3.5 - Facial features hard to recognise, due to the abstraction levels within cartooned frames. Therefore, in order to detect such emotions, one needs first to understand the background information about the actions taking place within the plot. Image taken from the iCartoonFace dataset [6].

being hard to recognise on a pattern level. Considering the fact that the signature of the artist is embedded within these drawings, such variations within different cartoons are often represented in a unique manner. Whilst this diversity is usually easily grasped by most human readers, autonomous systems often encounter difficulties when not trained on the specific feature alterations.

When it comes to the task of geo-localised translation, when a cartooned individual is modified to fit the cultural background of another region, one concern would be whether the emotions would be represented in identical manner in different societies. In "Culture and the Categorisation of Emotions" [27], Russell *et al.* raise the possibility that emotions can be perceived differently through the use of language, thus putting forward the possibility that human emotions are not universal, but vary due to the regional aspect of language. However, studies [28] [29] have shown that these lexical differences do not indicate any real variations in the way people perceive emotions, but rather a different manner in which they are verbally interpreted. One example would be the word *dor* in the Romanian language, which shows the strong desire to revisit a treasured place, or to see again someone or something dear to you. Another example would be the German word *schadenfreude*, which transmits the feeling of pleasure derived from another's displeasure. The fact that the English language does not include such words is not at all an indicator that English-speaking individuals are not capable of the above-mentioned emotions. Furthermore, it appears that only the so called complex emotions have such variations in perception within different regions of the world, and this is due to them being an aggregation of two or more emotional states. As a result, due to their



Figure 3.6 - Exemplification of the ambiguity presented within cartoons, when it comes to emotions. The above character's emotional state cannot be clearly categorised within the 7 universal emotions - neutral, happiness, sadness, anger, disgust, fear and surprise - due to the ability of cartoons to push the boundaries of the emotions which can be shared by humans through the use of facial expressions. Image taken from the iCartoonFace dataset [6].

complexity, they cannot be represented in a precise manner through facial expressions, but only through the use of language. The fact that this geo-localisation includes only the translation of images without text interpretation implies that the seven universal emotions which are being detected through the use of CNNs would not be regionally distinctive. One difference, however, is the fact that some societies consider it to be less acceptable to openly display strong negative emotions in social contexts. Therefore, as presented by Matsumoto *et al.* in [30], in some parts of the world the culture-specific rules of display make individuals suppress their emotions, lowering their expressiveness. However, this trend seems to not be consistent within the topic of cartoons, in which the characters show very suggestive behaviours regardless of these variations in expressiveness presented above.



Figure 3.7 - Exemplification of exaggerations within children's cartoons. Images taken from the "Tom and Jerry" cartoon [31] and from the iCartoonFace dataset [6].

In order to predict the emotion transmitted by a character, a rough estimation of the location of its face within the frame would be needed. In order to do so, I have first experimented with Haar Cascades, often used to detect the typical patterns found within human faces. Haar-like feature techniques are based on the fact that the aspect of humans contains structural similarities, despite the normal variety found within human faces. Therefore, these facial components are being searched within the frame, by applying Haar features in a manner similar to convolutional kernels. Examples of a Haar feature approach can be observed in *Figure 3.8*, where certain edge and line patterns are being searched for within the inputted frame, by computing a scalar product between the pixel values and the Haar templates [32]. However, because such techniques were initially developed such that cameras would be able to detect and thus auto-focus on faces, such algorithms only work for full-face images and often have issues detecting faces in a profile view. Therefore, even though Haar Cascades would be good for facial log-ins on personal phones, they lack robustness. The fact that cartoons often have characters facing different directions due to their interactions within the narrative plot, an algorithm based on the above-mentioned templates has achieved a 3.34% accuracy on the dataset containing 160 instances of cartooned characters, discussed about in *Section 2.2*. This low accuracy is also caused by the alterations of human features within cartoons. This is contrary to my initial thought, having believed that analysing Haar templates would be better suited for a cartooned facial detector, because facial features would be extracted separately rather than as a whole, so the positioning of features such as eyes, nose, etc within a face would be of less impact to the detector. However, the fact that cartoons do not present the normal shading found within humans' faces made the Haar Cascades unreliable.

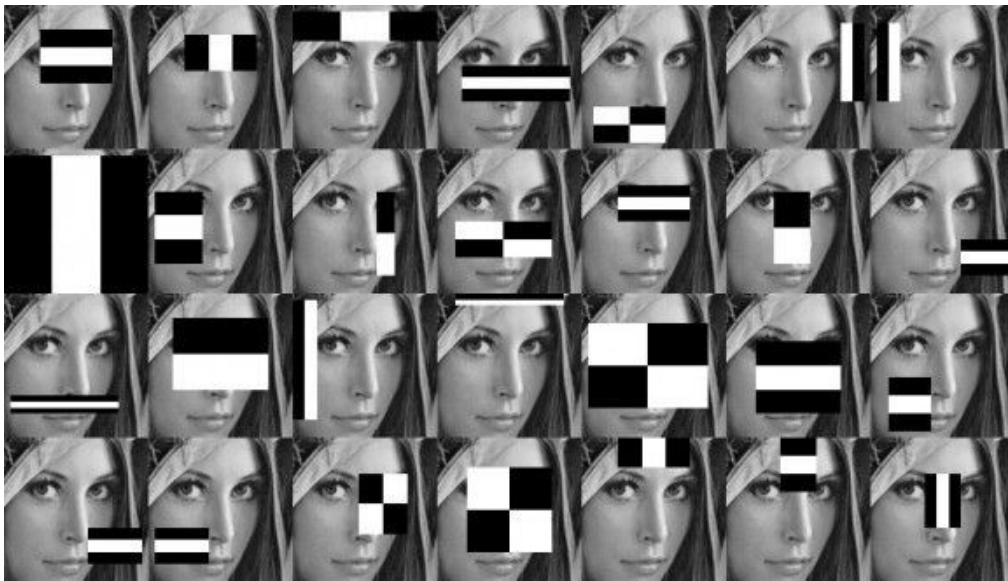


Figure 3.8 - Exemplification of a Haar feature technique, that looks for certain patterns within the human appearance in order to detect faces. The above technique is also known as the Viola-Jones Face Detection Algorithm [33]. The image was taken from [34].

After analysing the accuracy of the Haar feature approach, I have worked with different deep learning facial recognition systems, which achieve better results, despite not having been trained on cartooned characters. FER (Facial Expression Recognition) presented in [35] is an emotion detection library which uses a Multi-Cascade Convolutional Network to detect the faces within frames, alongside with their most probable expressed emotion. Likewise, Taigman *et al.* in [36] propose a model named DeepFace, which employs a nine-layer neural network with more than 120 million connection weights. The model performs the following sequential steps in order to predict the emotional state of a certain human instance: *detect*, *align*, *represent* and *classify*. Despite the fact that an accuracy of 97.35% is being achieved on frames depicting the real world as presented in [36], it occasionally encounters difficulties detecting very abstract human instances. However, it still achieves the best accuracy from all the approaches tested during the development of this project, being chosen to be utilised within the emotion module. Examples of well-detected cartoon emotions can be observed in *Figure 3.9*.

3.2.2 Gender and Age Analysis

Gender and age matching is yet another indicator that shows whether two human instances are compatible for replacement. Such analysis is crucial in order to keep the naturalness of the replacing frame and to maintain the coherence of the narrative thread throughout consecutive scenes. Noticeable discrepancy when it comes to the age or gender of a character would mean that the scene is no longer recognisable. However, a perfect match is not necessarily needed, as small variations in age would not modify the content of the frame. Therefore, due to the above-mentioned, I have concluded that a 80% similarity would be "good enough" when it comes to the age of two human instances within the context of geo-localisation.

As presented by Khan *et al.* in [37], geometric modeling techniques are the most widely used for gender detection, utilizing taught facial landmarks models to localise facial elements such as hair, nose or eyes. Such features are then further analysed in order to observe certain patterns which help categorize human faces as being either male or female. The same analysis is applied when determining a person's age, as ageing is a natural process which can be observed on any human being. However, such changes are not as pronounced within cartooned frames which makes it harder to predict a precise age, but can still determine a broader category in which the character should be placed, such as toddler or child.

The age and gender analyser is presented as a subsection of the above-mentioned facial emotion module, due to the fact that both parts extract meaning from exploring the human face. Whilst the body size is also an estimator of a person's age, a quantitative estimation is mainly derived from analysing the character's facial traits.



Figure 3.9 - Extracted emotions from several cartooned frames, with different artistic styles.

3.2.3 Body Position

It is often the case that cartooned characters present very specific body positions in order to anticipate movement, which would otherwise be difficult to express within motionless frames without pose exaggeration. Because of this, in order to design a precise image translation algorithm, one would also need to take the body position into account, as introducing elements which bring stillness to the frame or which do not match the movement presented by the initial character would change the aspect of the image. As a consequence, the context of the illustrations could be changed, thus no longer matching the presented events that take place along the narrative thread.

However, even though I have analysed the impact of body position within cartoons, I have not included the output within the human instance comparison algorithm as to not further complicate the translation task. When detecting the human pose within a frame, the positioning of the joints are being identified, which makes it easy to trace the skeleton of a person. By further analysing these points, Euclidean Distance can be used to determine the similarity between the detected poses, thus finding a better-suited character replacement. Examples of pose detection within illustrations can be found within *Figure 3.11*, the skeleton representation being presented on the left, whilst the detected joints mentioned above are being highlighted on the right side of the figure.

In order to detect such body positions, I have used the OpenPose architecture presented by Gupta *et al.* in [38]. The framework uses an 18-point model trained on the COCO dataset, which detects 18 pose-related landmarks such as neck, left shoulder, right shoulder, etc. For each one of such landmarks, a confidence map is being computed, which presents higher values at locations where the probability of a certain body part to be present is high. These heatmaps are being further analysed in order to detect the correlation between the identified key-points, thus determining whether they associate with the same detected human instance.

3.3 Matting and Compositing

Once segmented - by using the work of He *et al.* in [19] - new instances that belong to the same class as the original one but which match the new story location will be searched for. Through the use of matting [39], such instances are being estimated within their context and then extracted and blended with the original image through the use of the location of the detected segmentation masks. Due to the artistic factor of our project, this task would require extra steps, as the replaced items should be presented in an artistic form such that it would better integrate within the initial frame. For the non-cartooned replacing items only, I have been experimenting on using Neural Style Transfer and Image Cartoonisation to transform the replacing

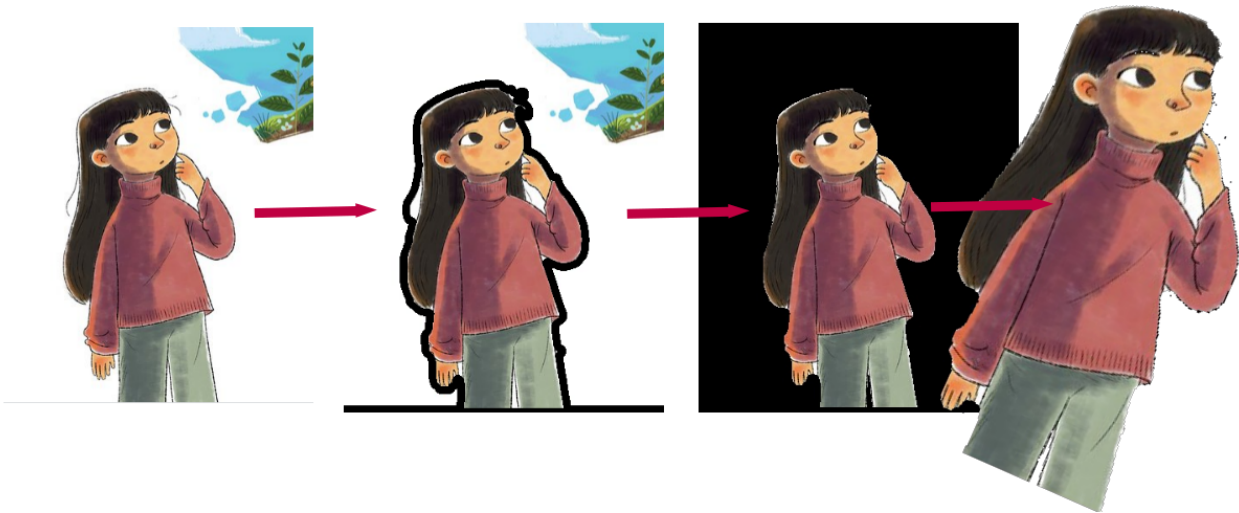


Figure 3.10 - Matting process, which extracts the replacing object from its frame, such that it can be positioned on the illustration based on the positioning of the segmentation mask.

objects, thus making each piece fit together and composing a more robust and natural experience (as better exemplified within *Subsections 2.2* and *2.3*). Previous work has been conducted on the interplay between the content and the style of arbitrary images [7], but has only been considered within the topic of art and has not been further used as a tool for compositing. Compositing techniques have been designed in order to combine visual elements from several sources such as if they belonged to the same scene. This process is a crucial part in image editing and cinematography, and started to be an important research topic within the Computer Vision community [40] [41].

In order to prepare an object for replacement, it first needs to be delimited from its background through the detection of hand-crafted features such as edges and corners. Once the object is well delimited, the background can be taken out by turning its pixels transparent. Such a process can be observed in *Figure 3.10*, where the extraction steps are being presented in sequence. Through the use of such an algorithm, the newly extracted human instance can be placed on top of the initial image at the location of the original object, without the background interfering with the rest of the image content.

In order to accurately place the object at the right location within the frame, the detected position of the segmented mask is being used, alongside with its width and height. Its shaping is used to detect the resizing factor that needs to be applied to the replacing object, as to make it better fit within the frame's context. The segmented mask is also used for compositing, as it is used to better mask the replaced object by modifying the mask area to a RGB value picked from its close vicinity.



Figure 3.11 - Open pose estimation technique, which detects the body shaping of a human instance by detecting its joints [38]



Figure 3.12 - Before and after object replacement.

PROJECT PLANNING

4.1 Project Research and Planning

After defining the project scope, researching and planning are two crucial aspects of the project management that would ensure a smoother progress throughout its development. When it comes to developing software, a thorough analysis of what has been done to date within the project's field is essential in order to understand the current capabilities of software systems, as well as to design a development plan based on real data. Having known what has been completed and how, provides with a head start which is necessary for any successful project. Planning can be conducted manually or through the use of tools designed for software development, as further discussed in the following section. When it comes to my Capstone Project, before the start of the first academic term, I have started researching in detail what has been done within the topic of Computer Vision related to image translation. After breaking down the project into smaller components which I viewed as necessary, such as emotion detection and object segmentation, I have taken an iterative approach in firstly researching and then planning each module of the project. The extensive summer preparations helped me have a more thorough understanding about my project's topic, but also about related topics such as the subject of emotional perception. Furthermore, during the summer period I have created a document entitled "Project Research and Planning" which contained detailed information about the project's motivations, expected development and proposed components, which can be found both on Jira (task B301288-9) and GitLab (within the "Written Content" folder). Such a document made it easier to revisit already read papers and to keep a more detailed written version of the project planning.

4.2 Jira

Agile development [42] was used during the whole duration of this project in order to bring structure to the work that needs to be completed. Agile development is a set of practices and frameworks used in software development which increase productivity and which are very helpful in flagging any issues encountered, as well as the risks that the project could potentially face. The main difference between Agile and other

software development techniques such as Waterfall is the fact that it brings a flexible and iterative approach to the project development, which helps to easily adapt to the changing aspect of a project. Therefore, due to its iterative approach, Agile minimizes the risks that can appear during the period of software development. This is also due to the fact that customers and stakeholders have the opportunity to observe the evolution of the project, which provides for easy feedback throughout the software development process.

Furthermore, the Agile methodology yields numerous benefits to its users through software management tools such as Jira, which have the purpose of monitoring the stages of development. Through storing data, Jira allows its users to retrieve and share information regarding certain tasks and to allocate parts of the workflow, as well as discuss about possible solutions to problems. Therefore, having stored information about already completed tasks, users can go back and analyze the way in which the problems encountered were overcome, thus helping them solve bugs more easily, by revisiting previously followed steps which have proved to be successful. Therefore, teams can collaborate more efficiently through the use of a common software. This approach also helps when it comes to remote working, as the users do not need to be in the same location in order to collaborate.

During the development of this project, the Jira software was used as the main tool for project management. It is a software which allows tickets to be created with a designated status - such as *task*, *story* or *epic* - while also being able to create workflows that can assist in the development of the project. The Kanban development tool is used as it provides more continuity to the project as opposed to a Scrum approach which is based on shorter and more concise sprints. Through the use of Kanban, one can always track the progress of a certain ticket, by moving it from the backlog to the different stages of development available: “Selected for Development”, “In Progress” and “Done” [43]. Therefore, Kanban uses cards containing tasks that can be moved around the board, allowing us to keep track of the status of the project by dividing its different components in different stages of development. In Jira, we can create five different types of cards, as presented below, which range from *tasks*, *stories*, *epics*, *bugs* and *risks*:

- A *task*, is an issue type in Jira that specifies a simple assignment that has to be completed.
- A *story* or so called “*user story*” is a software requirement that describes what has to be done using nontechnical language and it is usually split up in one or more tasks
- An *epic* represents a high-level view of a larger chunk of work within a project and it is often split in several stories and tasks
- A *bug*, is an issue that has been found during testing.
- A *risk*, is a possible issue that could be encountered during the project development.

Throughout the development of this project, there have been created several epics such as the "Contextual Object Similarity" Epic (B301288-88 in Jira), which are composed of several stories and finally tasks. Each such ticket created in Jira must have a distinctive name, that helps the user better understand the task required by their tickets. In some cases, when the title cannot be precise enough, a detailed description can be added in order to rule out any misunderstandings. Furthermore, each one of such ticket is being assigned a priority, a due date (if applicable) and can be linked to other tasks currently active within the workflow. Jira is a great tool which helped me keep track of all the tickets that I was currently working on in order to make sure no task was being left out. However, having worked mostly remotely, I found it difficult at times to access the above-mentioned tool, as CSEE Horizon was often out of action.

Finally, when completing a certain task, this action could be observed in the project workflow, being moved from "In Progress" to "Done" and attaching any relevant comments or files. Furthermore, when it comes to the code-related tasks, a link to the GitLab commit was attached, in order to link together the two tools used and to provide easy access to the newly created files or an easy view to the modifications made to already existing files.

4.3 GitLab

Another project management tool used in this project was GitLab, which is a Git-based software control platform that provides tools necessary for software development management. Git is a free and open source version control system designed to handle the software development process by providing coordination among programmers within the same team.

Such as Jira, GitLab is a great tool that helps teams better document their work, whilst being able to do so remotely. By using such platform, any team member is able to contribute and collaborate in a more straightforward manner, whilst ensuring the scripts are being held securely and privately. Besides the code, technical documentation can also be kept and updated within GitLab within a *README.md* file, which would be displayed within the project repository. Images can also be included within the documentation in order to better explain the concepts of the project, so anyone that reads it can understand what the project is about. Furthermore, the steps that need to be taken in order to use the software presented can be displayed, which would make the installation more straightforward, whilst ensuring the users are utilizing the right libraries and versions.

Through the use of Git, GitLab provides its users with the ability to track the changes produced to the files within the repository and to share such changes through a commit link. Furthermore, through the use of GitLab, one can easily modify the software without the risk of introducing bugs within the repository, as

these changes can be updated after the certainty that a working version of the software has been produced. Moreover, through the use of GitLab it is ensured that only authorised people are able to add, delete or modify content within the repository, as passwords or private keys are required to do so. Such tools also provide with a safe place to store your project data, which would not be lost due to computer breakdown and which can be retrieved from anywhere in the world.

All files relevant to my Capstone Project have been uploaded on the platform and can be found on my repository on GitLab. The technical documentation has been kept up to date and contains some of the important information regarding this project, as well as details about its use. I have also cloned my Conda environment and added it to the repository for the ease of installation, in order to make sure working versions of libraries are being used. Each time a modification has been made to one of the files, a commit message has been added and the commit was linked with the ticket number representing the task on Jira, as presented in *Figure 4.1*. By doing so, it is easier to get more information about the task and its completion.

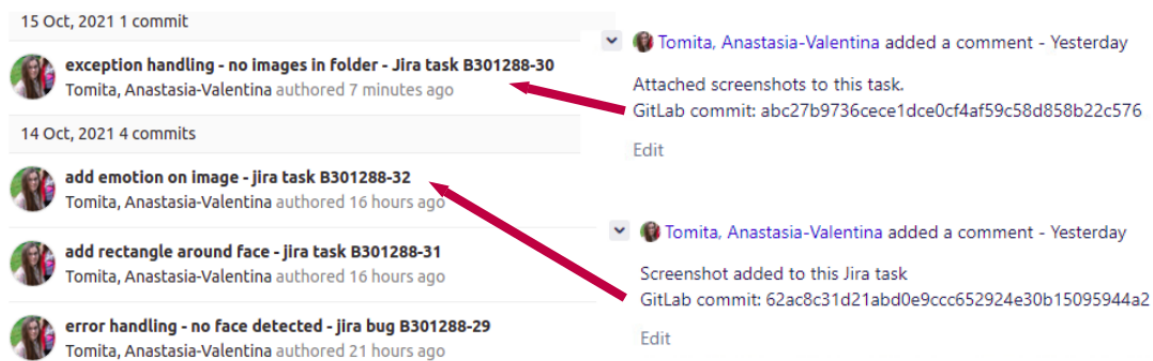


Figure 4.1 - Links made between the GitLab commits and Jira tasks/stories, which facilitate the linkage between the content presented by the two platforms.

RESULTS AND ANALYSIS

5.1 Analysis

The task of image geo-localised translation is not quite there yet, due to the limited amount of research undergone within the topic of children’s illustrations. There are several cases previously exemplified (*e.g.*: *Figures 3.5, 3.7 and 3.6*) which bring difficulties to any autonomous system, mostly due to the abstraction level found within illustrated frames. However, the possibility of such an algorithm to be achievable is definitely there, as CNNs are powerful enough to solve its vital component tasks (*e.g.*: *instance segmentation, emotion detection, etc.*) if being trained on the right amount of cartooned data. As believed within the AI community, “every task is one dataset away from being solved” (Unknown Author) and a well-built dataset would solve several of the challenges encountered during the development of this capstone project.

When it comes to the task of instance segmentation, which I believe to be the main component of the project, the results are promising. Even though some elements are being wrongfully detected, as observed in *Figure 3.2*, most objects are well segmented and well categorised. Further examples can be found in the Appendix. Supplementary training or fine tuning of the model would benefit the approach, where hand-segmented masks would be designed for a various range of cartooned frames, as observed in *Figure 5.1*, where I have annotated by hand each image pixels through the use of online segmentation tools [44].

However, when it comes to more detailed detection such as facial expression analysis or body position analysis, the accuracy is quite low. Haar-like features have achieved low results on cartooned frames, having a 3.34% accuracy on the dataset containing 160 instances of cartooned characters. On the other hand, deep learning techniques such as DeepFace achieve better accuracy on the same data, having a 16.87% accuracy on the above-mentioned dataset. However, such a percentage of correctly matched emotions is quite low for the algorithm to be reliable, needing a selection of images that contain characters which have features that better mimic human characteristics.



Figure 5.1 - Manually annotated image segmentation masks that I have designed for fine tuning.

CONCLUSION

6.1 Further Research

Foremost, due to the limited amount of cartoon-related training data which is openly available at the moment, an extensive dataset of illustrated characters from different locations in the world could be designed, which would contain characters having different facial expressions, various body positions and diverse clothing styles. In order to do so, a team of experts from different parts of the world would be needed, who would be able to accurately distinguish the cultural differences between such locations. By doing so, the cartoon style would remain consistent throughout the illustrations, which would bring a more natural feel to the image geo-localised translation.

Furthermore, within a certain book, it is often the case that in consecutive illustrations the same characters would be presented, but perhaps having different facial emotions, body positions or even clothing. This needs to be taken into consideration, as a new replacement with a distinct character would cause confusion to the reader. Because of this, through sequential frames the characters need to remain consistent, perhaps by assigning a certain identity to each detected human instance. This identity could be something simplistic such as a number, but which would remain consistent throughout the cartoon. In order to do this, one needs a detection module able to tell characters apart, as well as when another character enters the narrative thread. However, as mentioned above, facts such as emotion, body position and clothing would not be a good way to distinguish individuals, as they might not remain constant. On the other hand, elements such as age and gender should remain consistent throughout, thus being a conclusive determinant of the individual's identity, alongside other possible features.

6.2 Retrospection

This project definitely constituted my most important educational accomplishment to date, as it gave me the opportunity to be creative and to work on topics that I would have not covered otherwise. I have learnt a lot about the state-of-the art frameworks currently used within the Computer Vision community and the

challenges of working on an entirely new topic, where little or no help can be found online. I have also encountered the challenge of working with already built code whilst building upon it such that it would fit the task at hand. During my education so far, everything that I have built, either by myself or within a team, has been done from scratch. Because of this, I never had to understand large amounts of code or varied coding styles, neither did I have to deal with dependency issues, where one module would need a certain version of a library, whilst another module would need another. I have also learnt for the first time how to create and use Virtual Environments and to work remotely on a server through the use of Virtual Private Networks (VPN), fact which also allowed me to learn more about Ubuntu and its capabilities. I have also learnt about the current capabilities of computer algorithms within the topic of Computer Vision, whilst also better understanding their weaknesses and limits.

I am sure all of the above will benefit me in my future studies, as I now have a better understanding of how research is undertaken in academia, as well as the many challenges faced. This includes the increased volume of new information that is always changing, and working with more complex concepts that require building strong backgrounds in several intertwining disciplines, in this case being biology, psychology, cinematography and computer vision.

There have been a great amount of challenges throughout the project. Having worked on a topic that has not yet been covered, there has been a high level of uncertainty. The biggest challenge was that I have been working on cartoons, as opposed to pictures depicting the real world, and that there is very little work conducted on such illustrations. I have mitigated the issue related to cartooned datasets by styling and cartoon-ising normal images to bring a more cartooned feel to the frames. Moreover, in order to form small datastes myself, I have manually picked cartoon images from the internet, which also ended up being more challenging and time consuming than expected. However, I was still not able to further train the instance segmentation model, taking into consideration the project's time frame. As it was my first time trying to train a model, I encountered many issues when it came to inputting the data, due to the fact that all the tutorials showed that a JSON file (JavaScript Object Notation) is needed, but the Open Images Dataset comes with a CSV file without column annotations.

I consider that my Capstone Project definitely allowed me to put into practice everything that I have learnt so far whilst studying Computer Science, both during my High School years, as well as University. It has challenged my creative thinking and helped me further explore the field of Computer Science, especially when it comes to the state-of-the-art systems. Despite it being challenging or overwhelming at times, the project has given me an insight into how research is undertaken and what abilities are essential in order to do so. This includes the attention to detail needed to fully understand certain concepts, the independence needed to conduct your own research, and originality when it comes to the ability of extending previous

work as to solve new tasks. I believe that I have acquired essential skills during these months, which will definitely help me in my future career as a Computer Scientist.

6.3 Conclusion

Geo-localised translation is an exciting topic within the Computer Vision community, which would improve the quality of education in the most culturally diverse parts of the world. A very large proportion of the global population does not have access to education in their mother language, which has always brought the need of didactic content translation. However, the adaptation of illustrations is often complex, being dependant on the skills of a visual artist. Therefore, such a task is often overlooked, despite the fact that it would help young readers better relate to the stories depicted, by being able to associate the fictional universe created by the author with the world around them. For young readers, such animated books bring the means to interactively learn more about the way societies work, but also about human morality. Books also help children better understand and assimilate the language, whilst also building their emotional awareness.

REFERENCES

- [1] Kaushik Chavan, Roshan Bhoir, Nikhil Gawade, and Surekha Khot. Number/text translate from image. 2020.
- [2] Keechul Jung, Kwang In Kim, and Anil K Jain. Text information extraction in images and video: a survey. *Pattern recognition*, 37(5):977–997, 2004.
- [3] Pratham Books. Official website: Engaging storybooks in 21 indian languages. <https://prathambooks.org/>, April 2022.
- [4] Lu Jin, Shenghua Gao, Zechao Li, and Jinhui Tang. Hand-crafted features or machine learnt features? together they improve rgb-d object recognition. In *2014 IEEE International Symposium on Multimedia*, pages 311–319. IEEE, 2014.
- [5] Nikita Jain, Vedika Gupta, Shubham Shubham, Agam Madan, Ankit Chaudhary, and KC Santosh. Understanding cartoon emotion using integrated deep neural network on large dataset. *Neural Computing and Applications*, pages 1–21, 2021.
- [6] Yi Zheng, Yifan Zhao, Mengyuan Ren, He Yan, Xiangju Lu, Junhui Liu, and Jia Li. Cartoon face recognition: A benchmark dataset. In *Proceedings of the 28th ACM international conference on multimedia*, pages 2264–2272, 2020.
- [7] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. A neural algorithm of artistic style. *arXiv preprint arXiv:1508.06576*, 2015.
- [8] Yongcheng Jing, Yezhou Yang, Zunlei Feng, Jingwen Ye, Yizhou Yu, and Mingli Song. Neural style transfer: A review. *IEEE Transactions on Visualization and Computer Graphics*, 26(11):3365–3385, 2020.
- [9] Alexei A Efros and William T Freeman. Image quilting for texture synthesis and transfer. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 341–346, 2001.
- [10] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2414–2423, 2016.

- [11] Logan Engstrom. Fast style transfer. <https://github.com/lengstrom/fast-style-transfer>, March 2016.
- [12] Xinrui Wang and Jinze Yu. Learning to cartoonize using white-box cartoon representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8090–8099, 2020.
- [13] Yong Zhang, Weiming Dong, Oliver Deussen, Feiyue Huang, Ke Li, and Bao-Gang Hu. Data-driven face cartoon stylization. In *SIGGRAPH Asia 2014 Technical Briefs*, pages 1–4. 2014.
- [14] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Mallocci, Alexander Kolesnikov, et al. The open images dataset v4. *International Journal of Computer Vision*, 128(7):1956–1981, 2020.
- [15] Abdul Mueed Hafiz and Ghulam Mohiuddin Bhat. A survey on instance segmentation: state of the art. *International journal of multimedia information retrieval*, 9(3):171–189, 2020.
- [16] Alberto Garcia-Garcia, Sergio Orts-Escolano, Sergiu Oprea, Victor Villena-Martinez, and Jose Garcia-Rodriguez. A review on deep learning techniques applied to semantic segmentation. *arXiv preprint arXiv:1704.06857*, 2017.
- [17] Anurag Arnab and Philip HS Torr. Pixelwise instance segmentation with a dynamically instantiated network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 441–450, 2017.
- [18] Analytics Vidhya S. Pulkit. Computer vision tutorial: A step-by-step introduction to image segmentation techniques. <https://cdn.analyticsvidhya.com/wp-content/uploads/2019/03/Screenshot-from-2019-03-28-12-08-09.png>, April 2019.
- [19] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.
- [20] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015.
- [21] Saad Albawi, Tareq Abed Mohammed, and Saad Al-Zawi. Understanding of a convolutional neural network. In *2017 international conference on engineering and technology (ICET)*, pages 1–6. Ieee, 2017.
- [22] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.
- [23] Majid Rabbani and Paul W Jones. *Digital image compression techniques*, volume 7. SPIE press, 1991.

- [24] Dong Kyu Lee, Junyong In, and Sangseok Lee. Standard deviation and standard error of the mean. *Korean journal of anesthesiology*, 68(3):220, 2015.
- [25] Prudhvi Raj Dachapally. Facial emotion detection using convolutional neural networks and representational autoencoder units. *arXiv preprint arXiv:1706.01509*, 2017.
- [26] Abhishek Verma, Piyush Singh, and John Sahaya Rani Alex. Modified convolutional neural network architecture analysis for facial emotion recognition. In *2019 International Conference on Systems, Signals and Image Processing (IWSSIP)*, pages 169–173. IEEE, 2019.
- [27] James A Russell. Culture and the categorization of emotions. *Psychological bulletin*, 110(3):426, 1991.
- [28] David Matsumoto. Culture and emotion. 2001.
- [29] Dianne A Van Hemert, Ype H Poortinga, and Fons JR van de Vijver. Emotion and culture: A meta-analysis. *Cognition and emotion*, 21(5):913–943, 2007.
- [30] David Matsumoto and Paul Ekman. American-japanese cultural differences in intensity ratings of facial expressions of emotion. *Motivation and emotion*, 13(2):143–157, 1989.
- [31] Metro-Goldwyn-Mayer Studios. Tom and jerry cartoon. https://metro-goldwyn-mayer-cartoons.fandom.com/wiki/Tom_and_Jerry.
- [32] Yi-Qing Wang. An analysis of the viola-jones face detection algorithm. *Image Processing On Line*, 4:128–148, 2014.
- [33] Jamal M Al-Tuwaijari and Saja A Shaker. Face detection system based viola-jones algorithm. In *2020 6th International Engineering Conference “Sustainable Technology and Development”(IEC)*, pages 211–215. IEEE, 2020.
- [34] Analytics Vidhya Aditya Mittal. Haar cascades, explained. <https://medium.com/analytics-vidhya/haar-cascades-explained-38210e57970d>, December 2020.
- [35] Yousif Khairuddin and Zhuofa Chen. Facial emotion recognition: State of the art performance on fer2013. *arXiv preprint arXiv:2105.03588*, 2021.
- [36] Yaniv Taigman, Ming Yang, Marc’Aurelio Ranzato, and Lior Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1701–1708, 2014.
- [37] Khalil Khan, Muhammad Attique, Ikram Syed, and Asma Gul. Automatic gender classification through face segmentation. *Symmetry*, 11(6):770, 2019.
- [38] Vikas Gupta. Multi-person openpose. <https://learnopencv.com/multi-person-pose-estimation-in-opencv-using-openpose/>, September 2018.
- [39] Qiqi Hou and Feng Liu. Context-aware image matting for simultaneous foreground and alpha esti-

- mation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4130–4139, 2019.
- [40] Bor-Chun Chen and Andrew Kae. Toward realistic image compositing with adversarial learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8415–8424, 2019.
- [41] Fangneng Zhan, Shijian Lu, Changgong Zhang, Feiying Ma, and Xuansong Xie. Adversarial image composition with auxiliary illumination. In *Proceedings of the Asian Conference on Computer Vision*, 2020.
- [42] Martin Fowler, Jim Highsmith, et al. The agile manifesto. *Software development*, 9(8):28–35, 2001.
- [43] David Cohen, Mikael Lindvall, and Patricia Costa. Agile software development. *DACS SOAR Report*, 11:2003, 2003.
- [44] Segments.ai. Image segmentation tools. https://segments.ai/?campaignid=15205063410&adgroupid=136324721184&network=g&device=c&gclid=Cj0KCQjw06OTBhC_ARIsAAU1yOXg56BQL1zwqBxVb-QTz0QGpZDlhqoRum68JpZcFqV4HIAaAvUojOQwCB.