João Pedro Ramos Belmiro – 9791198
João Vitor Nasevicius Ramos – 9894540
Matheus Aparecido do Carmo Alves – 9791114
Victor Antonio de Oliveira– 9791326

ForeverPixelated: a skin generator for games.

Professor Doutor Moacir Antonelli Ponti

SCC0251 Image Processing

Instituto de Ciências Matemáticas e de Computação - ICMC
Universidade de São Paulo - USP
Bacharelado em Ciências de Computação

São Carlos, SP

July 13th, 2020

List of Figures

Contents

Abstract

Image processing and enhancement represents a significant challenge for modern computer science, once it can provide the opportunity to extract data and improve techniques for diverse purposes, e.g., fingerprint recognition, warehouse product cataloguing or the identification and classification of diseases. Besides these possibilities, image processing methods are in our daily life casual tasks, such as filtering photos on Instagram or modifying images by Photoshop. These usages are casual but demonstrate the image processing wide range and its incorporation in society. Games companies are producing games that enable content creation inside the game's universe. Animal Crossing: New Horizons, for example, allows the player to create outfits design (clothes or hats) and wear them. In this way, this project proposes the creation of a "skin generator" for some games that allows the design of content in-game via pixel-art. We propose to create the content's design through a local photo update and transform it into a pixelated design for the game, providing it to the user for casual purpose. Applying image processing techniques, we aim to present the result in an optimised and easy-to-use way.

Keywords: Image Processing; Image Segmentation; Image Features Extraction.

1 Introduction

1.1 Motivation and contextualization

Image processing and enhancement represents a relevant area for research and technological development in society while presenting significant challenges for modern computer science. These techniques are applied to solve distinct kinds of problems, including fingerprint recognition (1, 2, 3) , warehouse product cataloguing (4) or the identification and classification of diseases (5, 6, 7) . Besides the application, the usage of image processing methods can provide the opportunity to extract data from images that are “invisible” to the human eye. These invisible features can improve techniques for diverse purposes as presented and can change our daily life.

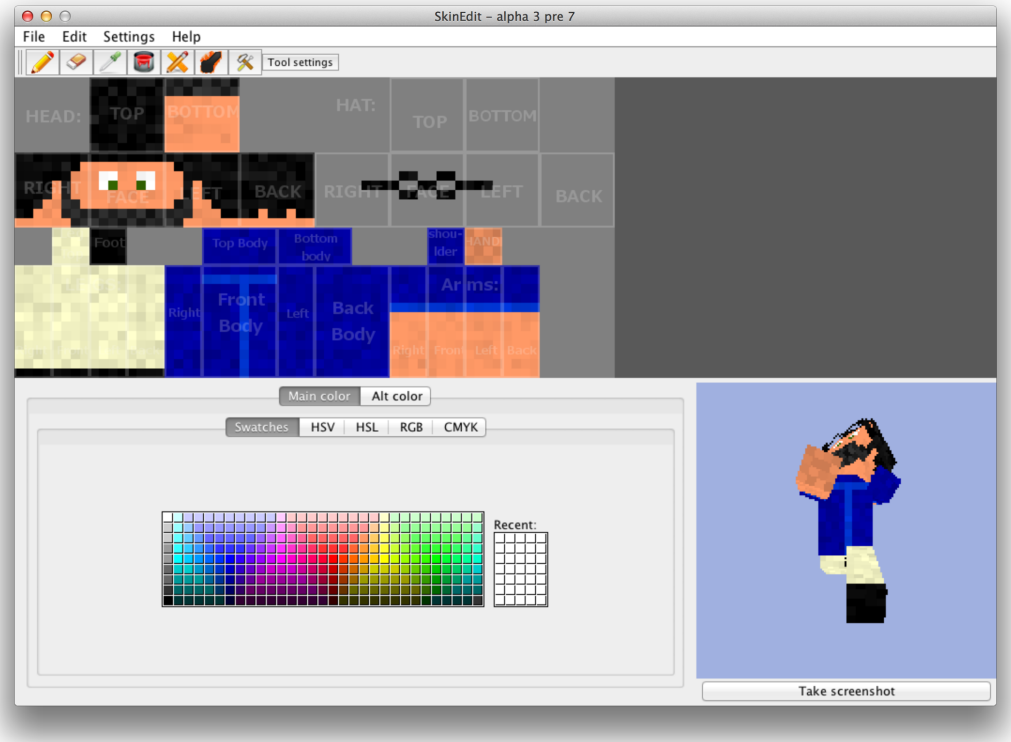
In our daily life, image processes are also performing casual tasks such as filtering photos on Instagram or modifying images on Photoshop. These usages are casual but demonstrate the image processing wide range and its incorporation in society. People are learning these methods and applying it casually in their purpose. They are creating new Instagram Filters, modifying images on-the-fly to innovate the content in their live streaming channels and erasing the inconvenient detail from memory photos.

Game companies are aware of this creativity and capability of creating the content of their players. So, companies are producing games that enable the creation of contents inside the game’s universe. *Super Mario Maker 2* is a World Maker mode that allows players to design a map and string multiple courses together in a Super Mario World-style. *Animal Crossing: New Horizons* allows players to create outfits design (clothes or hats) and wear them. *Minecraft* is a “create by yourself” game, where all can be changed and updated as you design and desire. Note that these three examples want and encourage players to create and modify its content as part of the gameplay experience.

In this way, this project proposes the creation of a "skin generator" for some games which allow the design of content in-game via pixel-art. Both presented games, *Animal Crossing* and *Minecraft*, enable the creation of character outfits, inside or outside the game. So, we will propose to solve the problem for these games outfit creation. Figure 1 presents the model used to import the outfits in-game.

We propose to create these contents design through a local photo update. The main challenge consists in transforming the local picture into a pixelated design for the game without losing the outfit main features. Applying image processing techniques, we aim to present the result in an optimised and easy-to-use way.

(a) Anima Crossing: New Horizons.



(b) Minecraft.

Figure 1 – Pixalated-outfit models for some games.

1.2 Objectives

In this document, the main objectives and the image processing we will apply to achieve them follow:

1. Extract the top wear from the uploaded local photo (Image Segmentation);
2. Remove the undesirable details from the resulting image (Image Contour and Pixel Color Modification);
3. Remove the perspective from the cleaned image (Perspective Transformation);
4. Transform the image into a pixelated design (Image Mapping), and;
5. Transform the pixelated image into the required game's format.

Additionally, all the implementation will be available at our public repository on GitHub[1], so the result will be free-to-use and without profit intend.

[1] Available at: <https://github.com/Micanga/skin_generator>

2 Methodology

Transforming real-world images into a pixelated model that fits a video-game model requests the application of diverse image processing techniques to achieve the final result. Given the difference between features that represent a real-world and a video-game model, we design a 5-step algorithm (related to the 5 objectives defined in Section 1.2) to approach the proposed problem and bring the world features closer to the game features. Therefore, this section aims to describe our approach and the methods used to reach the final objective. Section 2.1 will present our problem definition, describing input and output format and Section 2.2 will explain each step of our proposed algorithm.

2.1 Problem definition

The problem can be described as follow: Consider the input of 4 images that contain a recognisable person wearing any top-wear outfit into the front, back and sides perspectives. The problem consists of the top-wear extraction (in all perspectives) and its transformation into a game outfit model. For this problem, the output image would fit the *Animal Crossing: New Horizon* and *Minecraft* outfit model.

The input can be in any image format (e.g., JPEG, JPG, PNG) and do not need a user pre-processing, requiring just the correct upload of the perspectives. The Figure 2 represents a possible input for the problem.

(a) Front side. (b) Left side. (c) Back side. (d) Right side.

Figure 2 – Input example containing all perspectives of the outfit.

Note that the photos do not need to be centralised or follow any standard requirement to be an input to the program.

Given this brief definition, we must solve the problem using image processing techniques and present as result the output in the formats presented in Figure 1.

2.2 Approach and Partial Results

The methodology starts with the Fashion AI algorithm application, which is a blend of Image-Processing Techniques and Deep-Learning to segment and extracts a top-wear from a given image. As a brief explanation, the Fashion AI segmentation applies a *Deep Convolutional Neural Networks* to identify clothes, separating them from the background and removing the person standing in the picture. For this purpose, it uses *GrabCut*, an image segmentation method based on graph cuts to extract the outfit available in the picture. For further information, we recommend the read of Fashion-IA GitHub[1].

The result of this application is a two-layer image which shows the extracted top-wear outfit detached. In other words, the output from Fashion IA is a two-layer image that presents the original image overlaid by a mask deleting all information that does not represent a piece of clothing. However, after this process, some undesirable details remain in the image. Figure 3 presents the result from a clothing extraction by Fashion IA algorithm.

(a) Original Image.

(b) Fashion-IA Output.

Figure 3 – Output generated by Fashion-IA.

As Figure 3b presents, the clothing piece was extracted, but we have some features that does not represent it in the image. In this way, we decided to apply a cleaning up process to make the extraction more accurate, removing these "blurred objects".

For the cleaning up process, we design a 3-step algorithm that (a) transforms the image into a 1-layer figure, removing the transparent layer and colouring it as white; (b) contours the objects into the image, and; (c) removes the smaller contoured objects (smaller areas) that should not represent the clothes.

So, first, we need to separate the output mask from the original image. We perform a transparency removing process that checks the alpha byte of each pixel in the image. Basically, those pixels that are not fully opaque are coloured as white. Figure 4 shows the result from this process.

[1] Available at: <https://github.com/anish9/Fashion-AI-segmentation>

(a) Fashion-IA output. (b) Output.

Figure 4 – Transparency removing process output.

After this *intensity transformation*, we start to contour the resulting figure. The contouring has its focus only over significant areas, being able to identify and differentiate them via hierarchy analysis, where contours inside other contours are not drawn. We call this process as *Constrained Contouring*. Figure 5 presents a highlighted result of the contouring process.

(a) Fashion-IA output. (b) Contouring output. (c) Constrained output.

Figure 5 – Contouring process outputs.

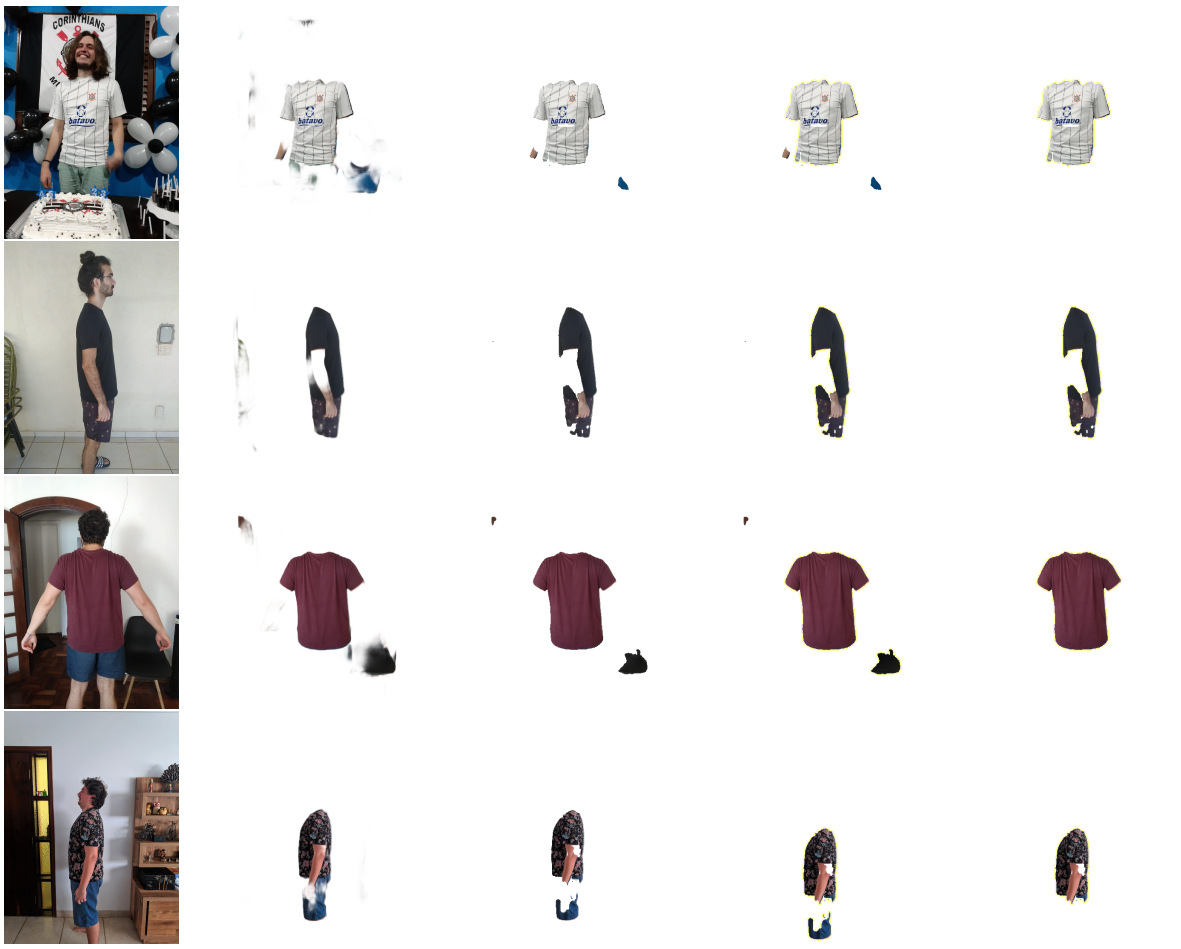
With the figure elements contoured, we start to remove the smaller outlined areas to maintain only the largest one. This process defines the last step of the cleaning up process. The whole process (cut, extraction and image cleaning) can be seen in Figure 6 for different given inputs to our algorithm.

The next step is the removal of perspective that enables the further mapping process. This process consists of a semi-manual procedure, which we select the image area via manual image-wrapping and remove the image perspective applying a geometric transformation. Specifically, we apply a warp transformation over the image content, deforming the pixel grid and mapping this deformed grid to the destination image. After the deformation, we obtain the desired perspective removal. The Figure 7 shows the highlighted step-by-step process.

Given the resulting image (with the perspective removed), we apply a simple filter to prepare the cloth for mapping. The filtering step consists of an image blurring followed

by a smoothing. Therefore, we apply a median blur method followed by a bilateral filter application. The idea is to blur the original image and smooth it to make easier the mapping process, from a real-world photo to a pixelated representation. Figure 8 presents the result after the filter process.

Finally, after the accomplishment of pre-processing, perspective removal and filtering processes, we start the image mapping to the game's model. At this point, we aim to get the processed photo and transform it into the game representation. So, the first step will focus on colour space reduction. As the games don't support the usage of full RGB space representation in a picture, we need to reduce the image's colour space to the game's colour space. In this way, we first apply an image resizing method, which interpolates the picture based on neighbours pixels area, and so we perform the *Mini Batch K-Means* technique that aims to reduce the figure colour space. At this point, we aim to guarantee here the correct dimensions and colouring to the image, enabling the quick upload to the game. The output is presented in Figure 9.



(a) Original Photo. (b) Extracted clothing piece. (c) Removing Transparency. (d) Contouring. (e) Cleaning up.

Figure 6 – Clothes extraction and pre-processing results.

(a) Original Image. (b) First perspective line. (c) Left-Top wrap.

(d) Left-Top-Right wrap. (e) Full perspective wrap. (f) Perspective removed.

Figure 7 – Removal perspective process (highlighted step-by-step).

(a) Before filtering. (b) After filtering.

Figure 8 – Filtering result.

With the final result picture, following this defined image processing procedure, the cloth must just be organised in the game's representation for upload and it will be ready to be a available content in game.

2.3 Pseudo-code

The complete procedure can be represented by the Algorithm 1.

Data: Array of top-wear clothes in all perspectives (front, back, left and right).
Result: Transformed real-world top-wear clothes into game's outfit model.

---

1. Apply the Fashion-IA algorithm;
2. Perform the clean up procedure:
for $twimage \in topwears$ **do**
  a. Remove the transparency;
  b. Contour the image;
  c. Remove the small areas;
end
3. Remove the perspective;
4. Filter the partial result;
5. Map to pixelated model;
for $twimage \in topwears$ **do**
  a. Resize the image;
  b. Reduce the colour space;
end
6. Return the game's model:

Algoritmo 1: Complete algorithm to extract and transform top-wear outfits into a pixelated model for games. The red text represents the future steps.

(a) Before resizing and colour reduction.

(b) After resizing and colour reduction.

Figure 9 – Resizing and Colour space reduction result.

3 Results

Applying the described method, we enabled the creation of games' content via image upload. Following the steps, we extract the cloth, pre-process, filter, map and transform the image to reach the result, presented in Figure 10.

(a) Original photo.

(b) In game-model.

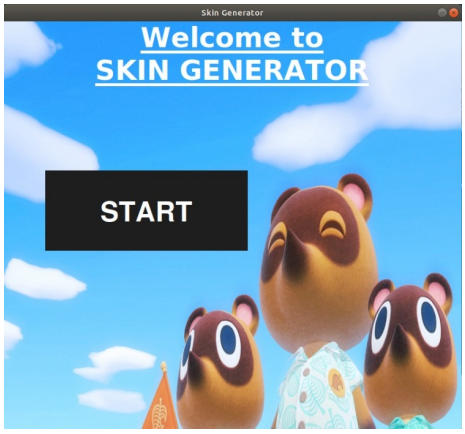
(c) Original photo.

(d) In game-model.

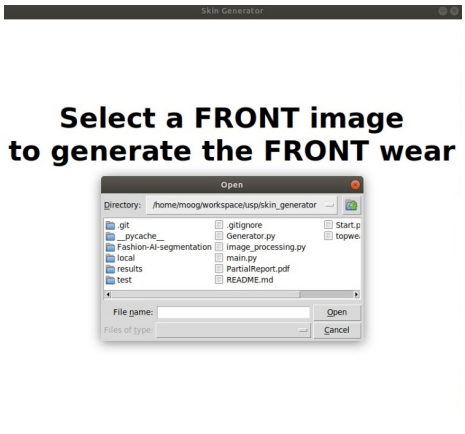
Figure 10 – Photo to Animal Crossing New Horizons' cloth.

As you can see, for *Animal Crossing: New Horizons*, we could create the photo's T-shirts models and use it at the game. Furthermore, it is possible to recognise relevant features present in the original cloths into the games cloths, fulfilling the principal concern and objective of this project.

In this way and facing the idea of proceeding this transformation easily, we implement and offer a simple platform to perform this image processing procedure. We present a simple platform with an intuitive interface that guides the user through the process. Figure 11 presents some platform interface examples.

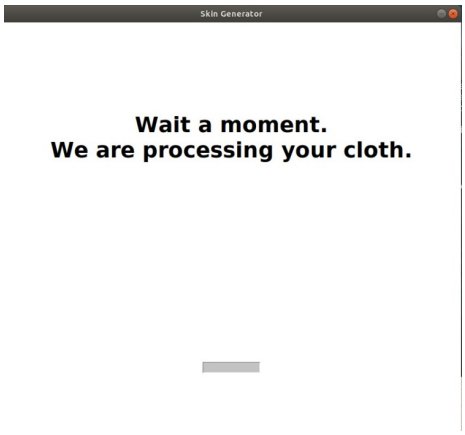


(a) Initial screen.

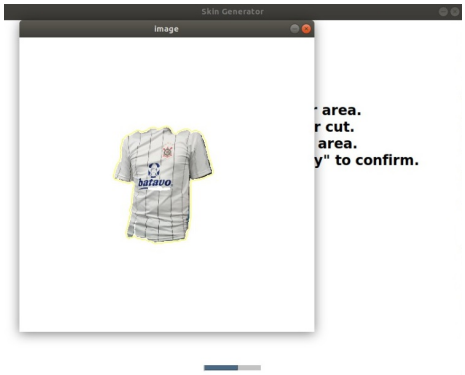


(b) Upload screen.

(c) Cloth type selection screen.



(d) Processing screen.



(e) Manual perspective removal screen.

Figure 11 – Interfaces of the ForeverPixelated platform.

We created each interface thinking to provide a friendly platform that requests minimal actions/interactions as possible to achieve the usage purpose. So basically the user needs to click in "'START", select the photo, cloth type and select the perspective area. Performing these three actions, the user must transform its photos into skins for the Animal Crossing game. For further information and explanation about the usage, check the project GitHub.

Finally, we tried to generate some of Minecraft's outfits using the same process. As the Minecraft is also a pixelated game which enables the creation and upload of outsource content into its world, we believe that the same method can fit it. Unfortunately, we could not get a copy of the game or test it in fact into the game. However, we reaffirm the reliability of the application of the *ForeverPixalated* method to generate "skins" for Minecraft.

In this way, we prospect to improve this functionality and the platform in future works, as well. The idea is to turn this casual tool available to the community, as explore image processing techniques.

4 Conclusion

Image processing and enhancement represent a relevant area for research and technological development, which can solve distinct kinds of problems. The usage of image processing methods can provide the opportunity to extract data from images and change our daily life. I.e., image processes are daily performing casual tasks such as filtering photos on Instagram or modifying images on Photoshop. These usages are casual but demonstrate the image processing wide range and its incorporation in society. People are learning these methods and applying it casually in their purpose. They are creating new Instagram Filters, modifying images on-the-fly to innovate the content in their live streaming channels and erasing the inconvenient detail from memory photos. Game companies are aware of this creativity and capability of creating the content of their players.

In this way, this project develops a "skin generator" for some games which allow the design of content in-game via pixel-art. We offer a platform which creates these contents design through a local photo update. The main challenge consists in transforming the local picture into a pixelated design for the game without losing the outfit main features. Applying a 6-step image processing technique, we aim extract, pre-process, remove perspective, filter, map and return the game's model to the user.

We believe this project could approach all the subjects presented during the module period, besides improve related knowledge and the students' capability to develop and apply image processing techniques. Furthermore, we achieve all the defined objectives for this proposal within the deadline and working all together.

Bibliography

1 ALI, M. M. et al. Overview of fingerprint recognition system. In: IEEE. *2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)*. [S.l.], 2016. p. 1334–1338. Citado na página 4.

2 ALSMIRAT, M. A. et al. Impact of digital fingerprint image quality on the fingerprint recognition accuracy. *Multimedia Tools and Applications*, Springer, v. 78, n. 3, p. 3649–3688, 2019. Citado na página 4.

3 CAO, K.; JAIN, A. K. Automated latent fingerprint recognition. *IEEE transactions on pattern analysis and machine intelligence*, IEEE, v. 41, n. 4, p. 788–800, 2018. Citado na página 4.

4 CANTRELL, R.; HIGH, D. R.; O'BRIEN, J. J. *Product inventorying using image differences*. [S.l.]: Google Patents, 2019. US Patent App. 16/257,297. Citado na página 4.

5 BENJUMEA, E. et al. Characterization of thermographic images of skin cancer lesions using digital image processing. In: IOP PUBLISHING. *Journal of Physics: Conference Series*. [S.l.], 2019. v. 1221, n. 1, p. 012076. Citado na página 4.

6 DHINGRA, G.; KUMAR, V.; JOSHI, H. D. A novel computer vision based neutrosophic approach for leaf disease identification and classification. *Measurement*, Elsevier, v. 135, p. 782–794, 2019. Citado na página 4.

7 KAVITHA, P.; PRABAKARAN, S. A novel hybrid segmentation method with particle swarm optimization and fuzzy c-mean based on partitioning the image for detecting lung cancer. Preprints, 2019. Citado na página 4.