1. Introduction

Our goal is "Remove the acnes in the picture automatically". Everyone wants to be beautiful, and the beauty filter has recently become more popular. So, we want to make a system on our own to remove the acnes in the picture or videos automatically. Figure 1 shows the main structure of our model.

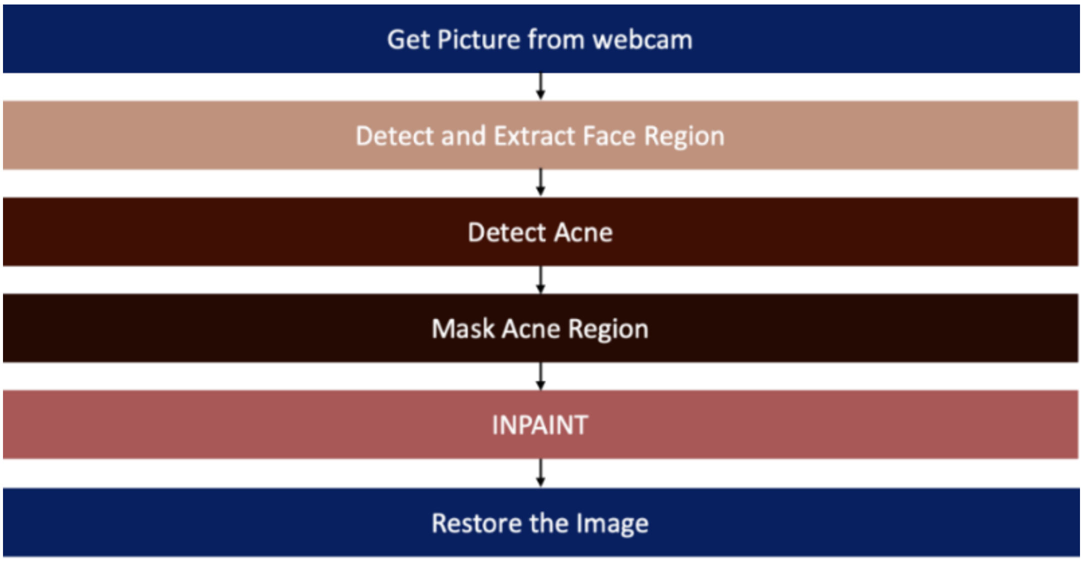

Figure 1 The flow chart of our model

2. Method

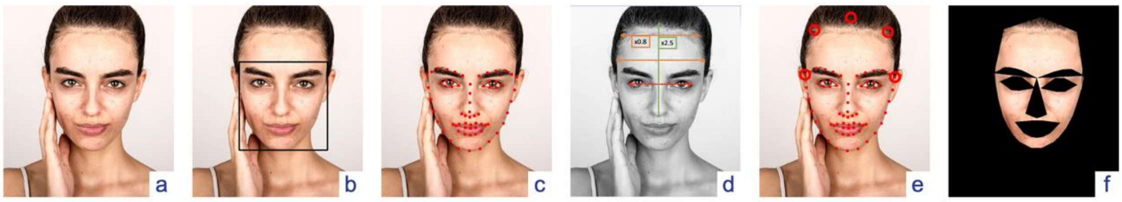

Figure 2 The pre-processing before detecting the acnes. (a) The figure is the original image. (b) The black rectangle contains the predicted face by Haar feature-based cascade classifiers. (c) The face features are predicted by shape_predictor_68_face_landmarks.

In the first step, we will identify the location of the people in the photo (Figure 2-a) and cut them out. The tool used is Haar feature-based cascade classifiers. The model requires a large number of photos containing faces and no faces for training. We can find the range of the face predicted by the classifier shown in Figure 2-b.

2.1. Find the landmarks on the face

Since we needed to remove the acne, we had to pick out the area precisely where the actual acne would appear. So, we have to get face features one step further.

Here we use the predictor model, **shape_predictor_68_face_landmarks.dat**, to detect the position of eyes, nose, mouth, etc. in the range of black rectangles containing faces. Figure 3-c shows the features with red points.

2.2. Find the forehead based on face features

However, the forehead information is not available in the prediction, mainly because of the fringe, which may lead to prediction errors. We use the predicted face features to calculate the position of the forehead. The red line is the connection between the two eyes.

The green line is 2.5 times the length of the nose and is perpendicular to the red line, and we find out the highest position of the forehead, the green point. The blue points are the origin face features. The orange line is parallel to the blue line and shifted up and scale down by a factor of 0.8, and we can find orange points lying on the contour of the forehead. This method can avoid a slight tilt of the face. Figure 3-d shows the lines and results.

2.3. Remain in the skin area on the face.

Finally, we hollowed out the eyes, nose, and mouth, leaving only the remaining skin area shown in figure 3-e and figure 3-f. Then we started the acne detection.

2.4. Detect Acne

After getting the skin picture, we can start detecting acnes.

In this part, we compared two methods from different papers. The first method we used is to compare the grayscale with the v channel in HSV to get ROI.(Figure 3)

In open cv, the grayscale uses the Y'CbCr 601 to convert, which is $0.2989 * R + 0.5870 * G + 0.1140 * B$, and the V in HSV is the max value of red, green and blue.

Because the face color stays between orange and yellow, and the acne color is red, so the grayscale of skin color will be larger than acne. So, we can detect the acnes in the picture by using v minus the grayscale of the skin picture.

After binarization, which means comparing the ROI with a constant threshold, we can get a result of the red area in the picture.

Due to not all of the red area are acnes, so we need to eliminate the false-positive by remove the area larger than a pre-defined area threshold.(Figure 4)

But the disadvantage of this method is that the thresholds are hard to define.

By the difference of the skin color, the area threshold will not be a constant. For example, in the original paper, the suggest threshold is 0.2, but in our picture, the best one is 0.13.

And for the area threshold, because the face size in the picture will not be the same, so it is not a constant, too.

Due to the lack of generalization ability, we can't use this method. We need another one.

Figure 3 The Red-difference Method

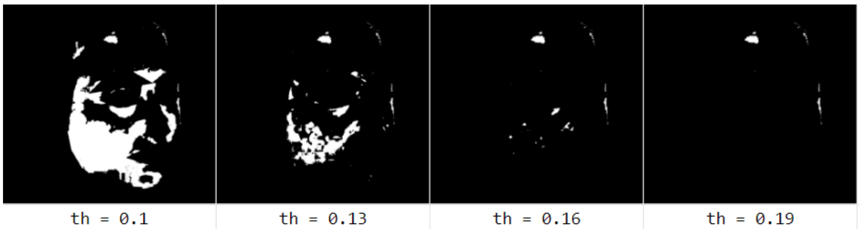

Figure 4 area threshold effects

The second method used the difference of area between face and acne. (Figure 5)

First, we define and get a picture A, which means the red value form the LAB color space. Because the LAB color space means the light, red-green and yellow-blue values, so the A here means the red-green color. Red goes to 1 and green goes to -1.

After get picture A, we can add a low-pass filter on it to remove the noise, which means the acnes. Because the acnes are small red area on the skin, it will not show in the low pass result.

So, we can get the ROI by doing A minus low pass of A. This way, we can get a more accurate acne mask.

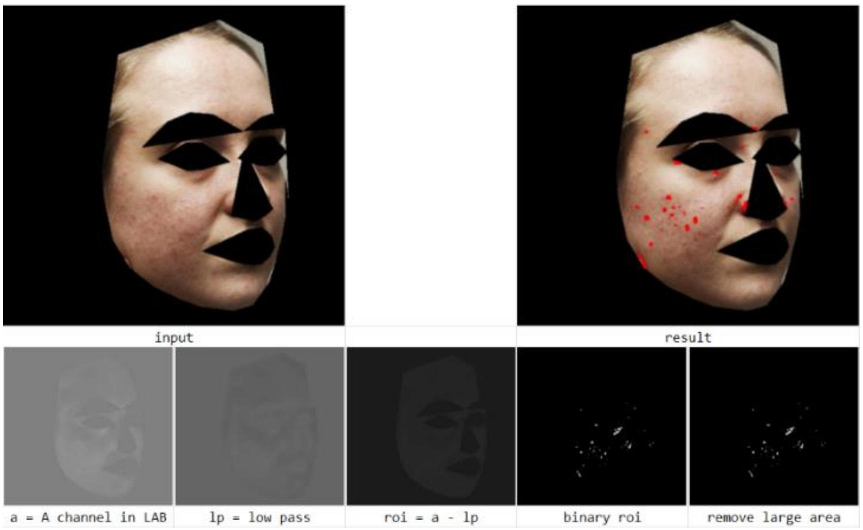

Figure 5 The Area-difference Method

2.5. Fast Matching Method

After getting the acne position, we can paint the correct skin color on them to remove the acnes. The first method to remove acne is Fast Matching Method(FMM), which used the openCV inpaint function.(Figure 7)

It will fill in the surrounding values to the area we set up the mask.

2.6. Deep Image Prior model

The second approach for inpainting is the deep image prior model (DIP), one of the ResNet models. The model proposes a different direction. It would train through one image only. And its application contains Denoise, Super Resolution, Inpainting tasks, etc. Here we use the inpainting application.

Figure 6 shows the frame structure of the Deep Prior model. Firstly, we will input the random values, Z, and pass them through layers such as downsampling, normalization, and other layers. In the middle of the process, it would start to upsample, normalize and restore to the original image. The result of downsampling will also directly pass as the input of upsampling layer.

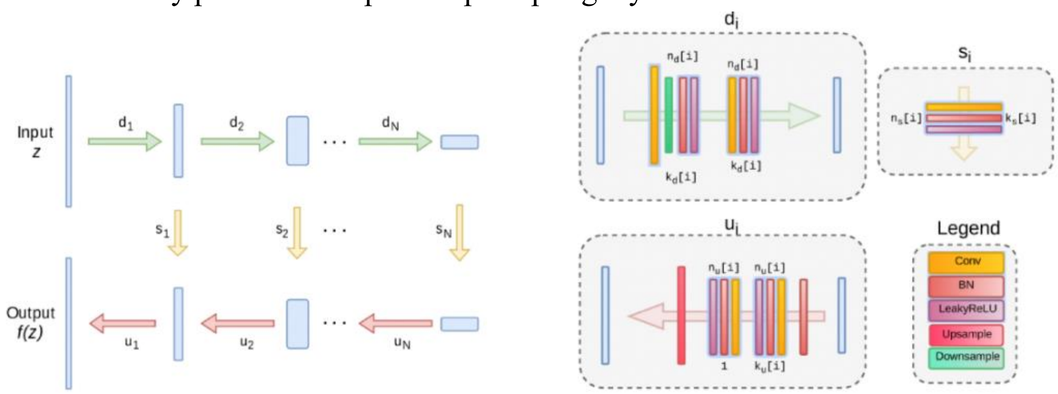

Figure 6 The structure of the Deep Image Prior model. The green arrows are downsampling. The red arrows are upsampling. And the yellow arrows are skip connections

The main idea is that we want the model to predict the image as the same as the original one without the acne area. Hence, In the loss function, we multiply the mask item on the predicted image and the original image to make the mean-square-error of the acne area become zero. The formula can be written as Equation 1.

$$E(x, x0) = ((x - x0) \odot m)^2, \text{ Equation (1)}$$

x is the predicted image, and x0 is the original image. m means the mask, containing two values 0 and 1. The value 0 presents the mask area in which the model should not learn by the original image. The value 1 is conversely.

3. Result

3.1. FMM method

The result with the FMM method is shown in Figure 7. With the naked eye, most of the acne has been moved.

Figure 7 FMM result. The left is the original image with the acnes. The right one is the restoration result by FMM.

3.2. DIP method

Figure 8 shows the recovery processing and final result. With the naked eye, most of the acne has been moved. The learning rate we set is 2e-5. The loss function is MSE. The optimizer is Adam. The number of epochs we set is 1800.

Figure 8 The recovery processing from left to right, from top to down. There are 18 images, and each image is predicted per 100 epochs.

3.3. Comparison with two methods

In comparison, the FMM has a problem that there will be errors at the edges of the face. Because the method just matches the mask with the nearest point, it may not knows what the shape of the face should be. However, using the DIP method, which uses machine learning, can predict the shape of the face to get a better result. (Figure 9)

Figure 9 Result comparison. The left image is the result of FMM. The right image is the result of the DIP model.

4. Conclusion

In conclusion, we have made a program that can remove the acnes on face. Also, we have compared several methods in the acne detection part and acne removing part. We found that while we using the low pass and DIP method, we can remove most of the acnes, and the contour of the face won't be weird. But the limitations are we can’t use the side-face as an input, and a large area of acne would affect the restoration result.

5. Reference

1. A. Bulat and G. Tzimiropoulos, "How far are we from solving the 2D & 3D face alignment problem? ", Proc. IEEE Int. Conf. Comput. Vis., 2017.
2. X. Wu, N. Wen, J. Liang, Y. Lai, D. She, M. Cheng, et al., "Joint Acne Image Grading and Counting via Label Distribution Learning", Proc. IEEE Int. Conf. Comput. Vis., 2019.
3. Y. Rejani and S. T. Selvi, "Early detection of breast cancer using SVM classifier technique", ArXiv Preprint ArXiv:0912.2314, 2009.
4. Kittigul, N. & Uyyanonvara, B. Automatic acne detection system for medical treatment progress report. In Information and Communication Technology for Embedded Systems (IC-ICTES), 2016 7th International Conference of, 41–44 (IEEE, 2016).
5. Chantharaphaichi, T., Uyyanonvara, B., Sinthanayothin, C. & Nishihara, A. Automatic acne detection for medical treatment. In Information and Communication Technology for Embedded Systems (IC-ICTES), 2015 6th International Conference of, 1–6 (IEEE, 2015).
6. G. O. Cula, P. R. Bargo and N. Kollias, "Imaging inflammatory acne: lesion detection and tracking", SPIE 7548 754801, 2010.Acne Detection with Deep Neural Networks
7. Kuladech Rashataprucksa, Chavalit Chuangchaichatchavarn, Sipat Triukose, Sirin Nitinawarat, Marisa Pongprutthipan, and Krerk Piromsopa. 2020. Acne Detection with Deep Neural Networks. In 2020 2nd International Conference on Image Processing and Machine Vision (IPMV 2020). Association for Computing Machinery, New York, NY, USA, 53–56.
8. Y. Yi, D. Qu and F. Xu, "Face Detection Method Based on Skin Color Segmentation and Eyes Verification," 2009 International Conference on Artificial Intelligence and Computational Intelligence, 2009, pp. 495-501, doi: 10.1109/AICI.2009.139.