AN ALGORITHM FOR AUTOMATIC SKIN SMOOTHING IN DIGITAL PORTRAITS

Changhyung Lee, Morgan T. Schramm**, Mireille Boutin*, Jan P. Allebach**

*School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN, 47907, USA
**Hewlett-Packard Company, 18110 SE 34th Street, Vancouver, WA, 98683, USA

ABSTRACT

We describe an automatic method for beautifying digital portraits by smoothing the skin of the face. The method builds on existing face detection and face feature alignment technology to automatically segment the face and neck areas to be smoothed. A smoothing filter is then applied to these areas. The resulting portraits are enhanced in a subtle and natural fashion.

1. INTRODUCTION

We present an automatic method for retouching digital portraits that aims to beautify faces by smoothing the skin in order to diminish wrinkles, blemishes, and other skin imperfections. Such retouching is routinely performed in the fashion and advertisement industries. But while some commercial applications currently exist to assist humans in this task, most of them still operate in a semi-automatic mode that requires user input to locate the face or the facial features. In contrast, our proposed method is completely automatic.

Our method builds on existing face detection and feature alignment technology to segment the relevant areas of the face. These areas are then used in combination with a skin segmentation to determine the region where smoothing is to be applied. There has been a lot of research on skin segmentation. Most of the current methods identify skin pixels based on color [1]. Our proposed skin segmentation method improves on such methods by using the face feature locations. More precisely, we create a skin map by segmenting the image using samples of the skin and non-skin data obtained from the aligned face features. We then model the skin and non-skin color distributions as Gaussian mixture models (GMM) and use a Bayesian segmentation algorithm to obtain the skin map. In other words, unlike the training-based approach that estimates the GMM from the training images in advance, we directly estimate the GMM by sampling the skin and non-skin data from the image being segmented. Hence this estimation adapts to each given face.

In the next section, we describe the face detector and feature extractor used in the first step of our method. Our proposed skin smoothing methodology is explained in Sect. 3. Some experimental results are presented in Sect. 4, and conclusions are given in Sect. 5.

2. FACE FEATURE LOCATION

The first step of our proposed portrait enhancement method is a preprocessing step in which all the faces contained in the images are located, along with their key features. This is done by first generating a bounding box around each face and subsequently identifying the location of the chin, nose, mouth, eyes, and eyebrows of each person. As a rough contour is sufficient for our purpose, a polygonal model is used to represent the feature contours.

This work was supported by the Hewlett-Packard Company.

Face detection is accomplished using a multi-view face detector based on the Viola-Jones technique [2], which relies on cascading classifiers in such a way that the classifiers in the early stage quickly discard simple non-face regions and the ones in the latter stage are designed to classify more complex cases. Each classifier is made up of a set of weak binary classifiers, which are learned from a training set following the AdaBoost algorithm.

Seven features are identified (the chin, mouth, nose, eyes, and eyebrows). Each feature is represented by a polygon. To align the features, we align the polygons using the active shape model method described in [3].

3. SKIN SMOOTHING

Skin smoothing must diminish the imperfections of the skin while preserving the important details of the face. We obtain this effect by delicately smearing the color of some strategic areas of the skin, following a manual procedure [4] commonly performed in graphics editing programs such as Adobe Photoshop. This procedure (Fig. 1(a)) comprises two steps. First, the whole image is smoothed out. Second, the image areas containing details that should not be smoothed (e.g. background, hair, beard, eyes, and nose) are restored. Blurring the image is accomplished by applying a diffusion filter to the entire image, and subsequently superposing the blurred image on top of the original one, setting the opacity to 50% so that both images are combined in equal amounts. The blurred image layer makes the skin look smoother, while the layer containing the original image preserves its natural appearance. However, the image areas containing details are also smoothed out by superposing the blurred image. To restore them, the blurred image is removed in these areas and replaced by the original image.

The skin map P_{skin}, whose construction will be discussed in the next section, takes on continuous values between 0 (non-skin) and 1 (skin). Let Y be the luminance channel of an original image. The filtered image Y_F is obtained by applying the smoothing filter F to Y, $Y_F = Y**F$, where $**$ denotes 2-D convolution. The image Y, the filtered image Y_F, and the skin map P_{skin} are combined as follows:

$$Y' = Y(1 - P_{skin}) + (\alpha Y_F + (1 - \alpha)Y)P_{skin}, \tag{1}$$

where Y' represents the output image and α represents the opacity factor. We use $\alpha = 0.5$. For smoothing, we use a Gaussian blurring filter with standard deviation $\sigma = H_f/100$, where H_f is the height of the face crop defined in Sect. 3.1, and only apply the filter to the luminance channel.

Fig. 1(b) summarizes the procedure for generating the skin map, which includes the skin of the face and neck area but not the facial details such as the mouth and eyes. We process each face individually. Using the feature point locations, we locate the skin and non-skin areas of the face and extract sample data of the color pixels of

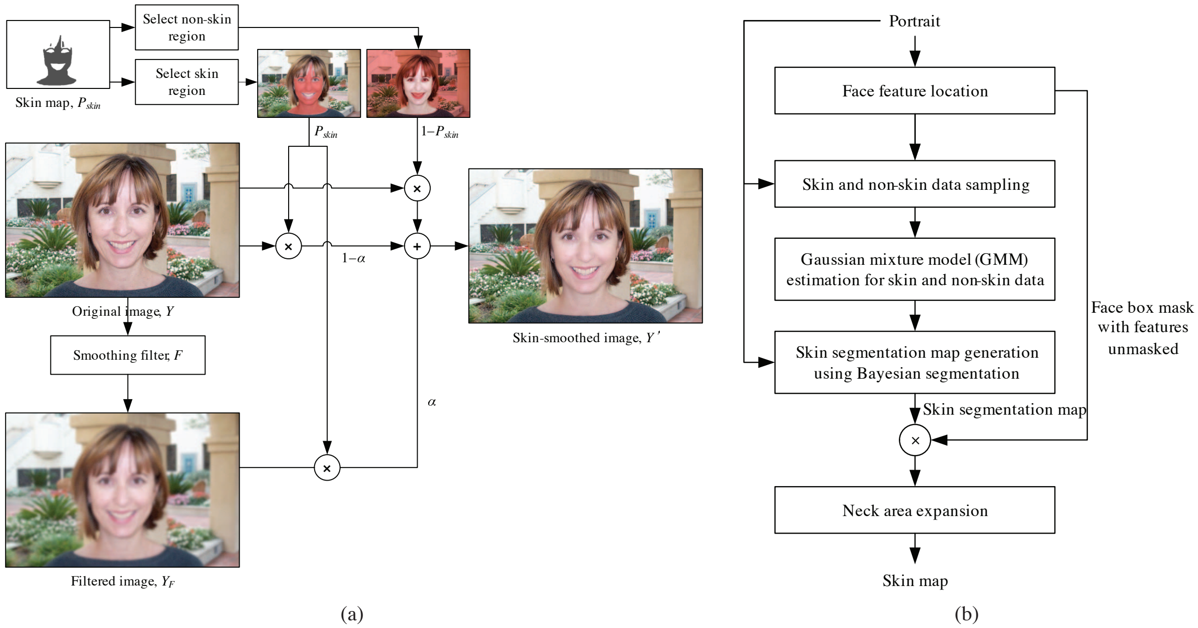

Fig. 1. (a) Smoothing framework. The filtered image and the original image are combined in the skin region. In the non-skin region, the original image is copied without any change. (b) Skin map generation framework.

the skin class and the non-skin class, as explained in Sect. 3.1 below. The data is then used to estimate the Gaussian mixture model (GMM) parameters for each class. Based on the GMM, we apply the Bayesian segmentation algorithm to the portrait to obtain the skin segmentation map. The details of this procedure are given in Sects. 3.2 and 3.3 below. Finally, the skin map is obtained by combining the skin segmentation map with the face box and subsequently extending the mask into the neck area, as described in Sect. 3.4.

3.1. Skin and Non-skin Data Sampling

The first step to obtain skin and non-skin data is to decimate the input image. Let H and W be the height and width of the input image, respectively. To reduce the processing time, we decimate the original image by an integer factor D if the image size is greater than $H_{\max} \times W_{\max}$, setting $D = \lceil \max(\frac{H}{H_{\max}}, \frac{W}{W_{\max}}) \rceil$. The height H_d and the width W_d of the decimated image are given by $H_d = \frac{H}{D}$, and $W_d = \frac{W}{D}$. We use $H_{\max} = 900$ and $W_{\max} = 1200$.

The next step is to crop out the face. In order to collect the skin and non-skin data from the face crop, the crop must include not only the face box but also the hair and the background near the face. The face box obtained by the face detector is not adequate for this purpose, so we define a new face box. The new box is calculated based on four extreme feature points, $\boldsymbol{p}_L = (x_L, y_L)$, $\boldsymbol{p}_R = (x_R, y_R)$, $\boldsymbol{p}_T = (x_T, y_T)$, and $\boldsymbol{p}_B = (x_B, y_B)$ located at the tips of eyebrows, top of the nose, and bottom of the chin, respectively as shown in Fig. 2(a). The width of the initial face box from the feature points is $|x_L - x_R|$, and the height is $|y_B - y_T|$. Then we add 20% left and right margins and 110% and 10% top and bottom margins to the initial face box, respectively. These margins are set so to include the whole face and hair regions in most images and to include as little background area as possible.. The face box is shown with a red contour in Fig. 2(a).

It is important to select the skin samples from an appropriate region of the face. This can be interpreted as creating a skin mask covering some areas of the face. This skin mask is constructed so to enclose most of the cheek area. To create the skin mask, we draw a polygon using the feature points to crop out the lower facial region. The polygon is drawn to exclude the mouth and nostril regions so that moustaches and beards—if present—are not included in the mask. The area inside the polygon is set to 255, and the other area is set to 0. The mask is then eroded by first blurring the mask and then setting the nonzero values smaller than 255 to 0. The eroded mask covers only the inner facial area and avoids the skin/non-skin boundaries. In general, sampling the data from this inner region improves the accuracy of the parameter estimation. The skin mask and the area marked by the mask are shown in Figs. 2(b) and (c).

The non-skin mask is constructed by excluding the face and neck from the face box. First we estimate the face ellipse using the feature points on the face boundary. Since we only have feature points on the lower face boundary, we add another point $\boldsymbol{p}_H = (x_B, y_T - 0.8|y_B - y_T|)$ which roughly falls onto the top of the head of the face to make the estimation more robust. Using the face boundary points and $\boldsymbol{p}_H$, we estimate the ellipse that best fits the points in the least-square sense. The neck area is estimated as a rectangle that is cut by the face box as shown in Fig. 2(d). The one side of the rectangle is a line that lies on the short axis of the ellipse. The length of the line is set to 80% of the short axis. The two other adjoining sides of the rectangle start from the end points of the line on the short axis and traverse until the lines are cut by the face box. They are parallel to the long axis of the ellipse. The fourth side of the rectangle coincides with the lower boundary of the face box. After excluding the face ellipse and the neck area from the face box, we add back the facial feature areas to the non-skin mask. This addition improves the ability to segment the non-skin regions inside the face. The non-skin mask and the area marked by the non-skin mask are shown in Figs. 2(d) and (e), respectively. After locating the skin and non-skin regions, we collect data samples from each region as input

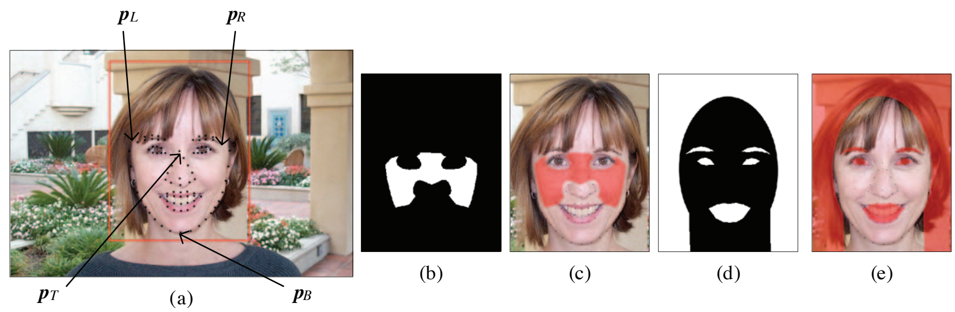

Fig. 2. Skin and non-skin mask creation. (a) Face box and feature points. (b) Skin mask. (c) Area marked by the skin mask. (d) Non-skin mask. (e) Area marked by the non-skin mask.

data for GMM estimation.

3.2. Gaussian Mixture Model Estimation

After sampling the skin and non-skin data from the input image, we model the pixel color distribution (in RGB format) for each type of data as a Gaussian mixture model (GMM). To do this, we need to estimate the parameters of the two GMMs, i.e. the number of subclasses K and the parameters for each of the subclasses. We do this using the maximum likelihood estimator following the approach described in [5], which uses an expectation-maximization algorithm and a sequential agglomerative clustering to optimize a criterion suggested by Rissanen called the *minimum description length*.

The input to this procedure is a vector of three-dimensional color pixel values whose length is restricted to 10^3 by subsampling. The initial K is set to 20. For most test images, the final estimated K for both skin and non-skin GMMs is much smaller than 20. For example, in the case of the picture shown in Fig. 2, we have $K = 3$ for the skin class and $K = 8$ for the non-skin class.

3.3. Skin Area Segmentation

Using the estimated GMM, we segment the portrait into skin and non-skin regions by applying a Bayesian segmentation algorithm. One approach to solve the Bayesian segmentation problem is to model the label of each pixel as a Markov random field (MRF) and to segment the image by approximately computing the maximum *a posteriori* (MAP) estimate of the pixel labels. We do this using a multiscale random field (MSRF) in conjunction with a sequential MAP (SMAP) estimator following the method described in [6]. This method replaces the MRF model with the MSRF model and the MAP estimator with the SMAP estimator. The MSRF model is a series of random fields progressing from coarse to fine scale. The series of fields are stacked like a pyramid and form a Markov chain in scale. The SMAP estimator minimizes the expected size of the largest misclassified region by assigning progressively larger cost to errors at coarser scales of the random fields. Unlike the MAP estimator, the SMAP estimator is not iterative and can be computed in time proportional to MN, where M is the number of classes and N is the number of pixels. In our case, $M = 2$ and $N < H_{\max} \times W_{\max} = 900 \times 1200$.

Figure 3 shows two skin segmentation maps computed using this approach. For both portraits, skin regions are identified with high accuracy. In Fig. 3(a), the hair region is particularly well segmented. However, some false positives are shown in the teeth region and the background. The false positives in the background will be removed in the final face skin map by combining the skin segmentation map with the face box. In Fig. 3(b), the beard region is especially well classified as non-skin; however, some portion of the moustache is falsely classified as skin and some portion of the neck area is missing. But small misclassified regions do not produce highly noticeable effects in the final skin smoothed images.

Fig. 3. Skin segmentation map.

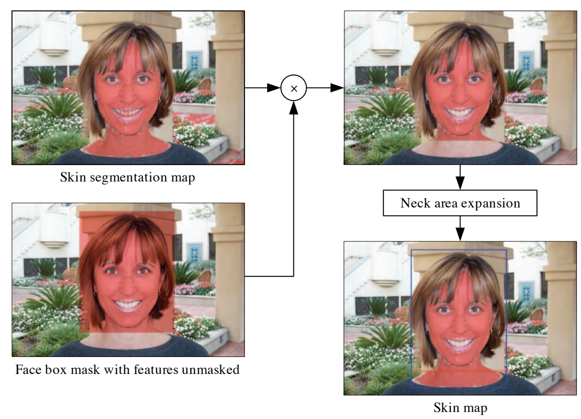

Fig. 4. Skin map generation from skin segmentation map and face box mask with features unmasked.

3.4. Skin Map Generation

To create the final skin map, we combine the skin segmentation map and a face box mask with face features unmasked. The face skin is then expanded to the neck area (Fig. 4). By combining the skin segmentation map and the face box mask, the false positives outside the face region are removed. As the face features in the face box mask are unmasked, the false skin positives on the features are removed.

To recover the skin regions cut by the the face box mask in the neck area, the connected components algorithm is applied to each pixel on the bottom side line of the face box on the skin segmentation map. In Fig. 4, the red line shows the bottom side line of the initial face box, and the blue line shows the bottom side line after the neck area expansion.

4. EXPERIMENTAL RESULTS

We implemented our algorithm in C and C++ and applied it to the images of the Caltech face database [7]. This database consists of 450 upright, frontal face images of size 592×896 pixels taken under various illumination conditions and backgrounds. Figure 5 shows some of our results. As one can see, the skin is smoothed out in a natural fashion while the facial feature details are preserved. In particular, flecks, blemishes, and wrinkles on the faces are greatly reduced. Some minor undesirable effects are created due to some slight segmentation errors in the face skin map. For example, in the case of the first image, there is some sharpness loss in the moustache area due to some pixels falsely classified as skin, as shown in the face skin map in Fig. 3(b). But these undesirable effects are hard to notice without a detailed comparison with the original image.

Without optimizing for speed, the average running time of our skin smoothing algorithm was 3.6 seconds once the face feature locations were found (on a PC with a single Pentium 4 2.8 GHz CPU and 1.5 G of memory).

5. CONCLUSION

We have presented a method for automatically enhancing portraits by smoothing the skin. This method achieves results that are similar to those obtained by hand using an image editing tool. The enhanced portrait are visibly better, as wrinkles, blemishes and other skin imperfections are significantly reduced. This shows that combining low level segmentation methods with some high level semantic information allows one to automate complex manual digitial editing tasks.

6. REFERENCES

[1] S. L. Phung, A. Bouzerdoum, and D. Chai, "Skin segmentation using color pixel classification: analysis and comparison," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 27, no. 1, pp. 148–154, Jan 2005.

[2] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2001, pp. 511–518.

[3] L. Zhang, H. Ai, S. Xin, C. Huang, S. Tsukiji, and S. Lao, "Robust face alignment based on local texture classifiers," in *Proc. IEEE Int'l Conf. Image Processing*, 2005, pp. 354–357.

[4] B. Jackson, *Photoshop cosmetic surgery*, Lark books, New York, 2006.

[5] Charles A. Bouman, "CLUSTER: an unsupervised algorithm for modeling Gaussian mixtures," `https://engineering.purdue.edu/~bouman/software/cluster/manual.pdf`.

[6] C. A. Bouman and M. Shapiro, "A multiscale random field model for Bayesian image segmentation," *IEEE Trans. Image Processing*, vol. 3, no. 2, pp. 162–177, March 1994.

[7] "Caltech faces 1999 (front)," `http://www.vision.caltech.edu/html-files/archive.html`.

Fig. 5. Sample results: the skin-smoothed images (second and fourth) compare favorably to the original images (first and third).