Contents

List of Figures

List of Tables

1 Introduction

We often cannot judge based on the appearance of a product worn by a model how that product will actually look on ourselves. Compared to users, the model may have very different skin colour and other bodily features. Ideally, to help the user make a better informed purchasing decision, the model demonstrating the product should be customized to each user. To achieve this, there are already many applications developed for the virtual trying on of products in the beauty, cosmetics and garment industries which digitally modify the images of models to take on the appearance of the user [9] [10, 11].

For this project, we have a video in a mobile app demonstrating nail polish on a model hand to allow virtual try-on of different nail polish colours. We would like to edit this demo video so that the model appears to have the user's exact skin colour, to help the user better determine whether the nail polish colours look pleasant on the user's own hand. While it is possible to manually prepare a series of demo videos with models of a range skin colours, preparing even a single video for virtual try-on is an extremely time intensive task. Moreover, each person has a particular skin colour and it's preferable to be able to tailor the video exactly to the skin colour of the user while the user is using the app.

To address these challenges, we propose developing an algorithm to incorporate into the app that quickly and automatically performs the image editing task while the user is using the app. The user should be able to provide an image of their own hand as input, and the video of the model hand should be convincingly and accurately modified to having the skin colour of the provided user image. A wide range of user skin colours should be supported by a single model of mid-toned skin, and the process should be able to run quickly on a mobile device, such that the user notices no significant time lag to see the resulting video frames upon inputting their own skin colour.

Currently, we aren't aware of an existing algorithm that satisfies all our specific requirements. While there has been a large body of work done addressing transfer of colour between images in general [4, 12–15], only a smaller set of work specifically addresses transfer of skin colour [6–8]. All such studies address face skin colour rather than hand skin colour, which often means that more of the study is devoted to handling colour transfer of different, more complex aspects of the face [7]. Skin colour transfer is also used as parts of other, more general imaging processing applications, but in those cases, since the skin colour transfer is often only a small part of the whole process, the algorithm used is often relatively simple and not heavily designed for achieving accuracy to the user skin colour [10, 11]. In the related field of skin colour enhancement applications, the methods used often are not meant to make large changes to the user skin colour [16, 17]. Finally, algorithms developed by most of the prior studies do not appear to be meant for use with the limited resources on a mobile device. We discuss these previous studies of methods of skin colour transfer in detail in section 2.2.

1.1 The goals, constraints and requirements for an effective skin colour transfer algorithm

Our project is intended to manipulate image frames in a video of a model hand demonstrating nail polish product so that the model hand takes on the user's skin colour. The images we must process will mostly consist of the back of a single hand shown prominently in the image. We expect image sizes the algorithm should be able to handle to be approximately 800 x 800 px and larger. We show an example of the desired output of our algorithm in Table 1.

Table 1: Example of our desired result given an original (the model) and a target image (from the user)

No.	Original	Target	Result
1.1			

To narrow the scope of our project, we will not include skin detection as part of this project and assume that our algorithm is already given a mask of the skin areas of all the images. We will focus solely on the transfer of the hand skin colour.

Based on our goals and the nature of our project, we list below several constraints and design paradigms against which we will evaluate our algorithm:

Compatibile with mobile device: Our algorithm is ultimately intended to support and application on a mobile device, so we must ensure that our code is portable to mobile devices and that the algorithm we develop can operate quickly with the limited resources of a mobile device so that the user will be able to see near-instant results.

Fully automatic: Since the goal of our project is for a commercial user to be able to change the model to his or her own skin colour, our algorithm cannot rely on any user input to perform the image editing and should able to accept only an image containing the user's own hand as the target image to transfer the colour of the model hand to.

Realistic skin colour transfer: Since the goal of our project is to perform skin colour transfer for model images that are meant to demonstrate cosmetic products to users, and the results are meant to invoke for the user the impression that the user's own hand is wearing the product, our final images must look as realistic as possible to avoid a displeasing, uncanny valley effect. Furthermore, the images we process will be large and feature a the skin on the back of a hand very prominently, so we can expect the realistic appearance we can expect that our result will be very heavily scrutinized by the user.

Accurate skin colour transfer: Since the entire goal of nail polish try-on application is to demonstrate to the user how a particular shade of nail polish will appear on his or her own hand, we must ensure that the results of the algorithm, more than looking pleasing to the user, actually matches the skin colour sample provided by the user exactly.

Wide range of colour transfer: Since the goal of the project is to reduce the number of nail polish try-on videos of different skin colours needed and since users may have a wide range of skin colour and should all be supported, our algorithm needs to be able to transfer the skin colour of a mid-toned hand to as wide a range of skin colour as possible.

2 Background and Literature Review

2.1 Changing and matching skin colour in Photoshop

Skin colour correction is a frequent problem encountered in photo retouching and there are a wide range of online video tutorials available documenting the methods artists use to manually adjust human skin tone in individual images using Adobe Photoshop, a widely used commercial image manipulation software. The purposes of these videos include giving the subject of an image the appearance of a tan, matching the skin tone of the subject to a desired skin tone on another individual, or matching the skin tone of a subject's face to the rest of the subject's body, which is often a slightly different colour [2, 3, 18]. Bearing in mind that techniques described by such tutorials expect artistic input from a human editor to acheive the results and are therefore not entirely aligned with the purposes of this project, it is useful to study these methods because the results achieved are usually extremely realistic and aesthetically pleasing and should be a standard that the algorithm developed in this project strives towards. We therefore surveyed a number of these videos and summarize below the techniques of some of the most relevant videos.

Summary of Photoshop techniques

Shaver demonstrated how to change a person's skin colour from dark to light [1], which is an impressively wide range to change. Shaver used levels and curves, which are tools that manipulate the rgb colour histogram of the image, to increase brightness to an extent, then performed further brightening by using a grey scale conversion to brighten the skin area of a black and white image and then using the luminosity blend mode to place the colour back into the image. We show the results acheived in Table 2.

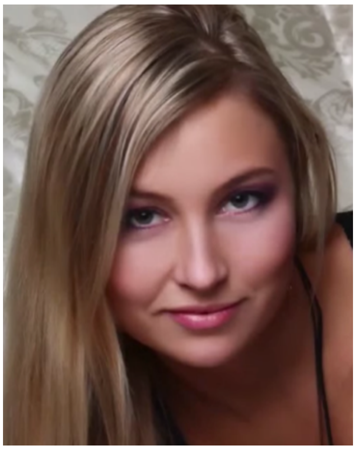
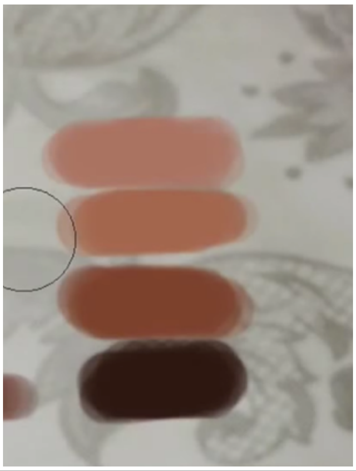
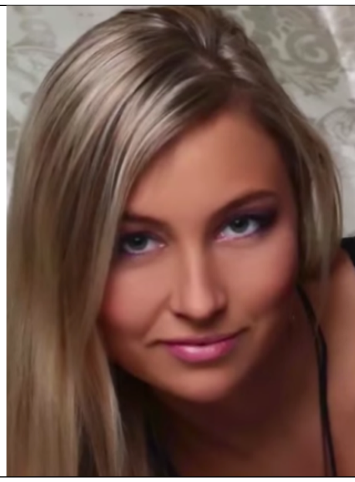
Table 2: Screen captures from Photoshop tutorial for changing skin colour from dark to light.

Original	Result
	

Phlearn demonstrated an effect in the reverse direction by demonstrating a technique for giving the model the image appearance of a dark tan [18]. The highlights and shadows of the image are adjusted separately by using the "blend if" function of Photoshop, which blends in an effect only if the original pixel is above a certain threshold of brightness.

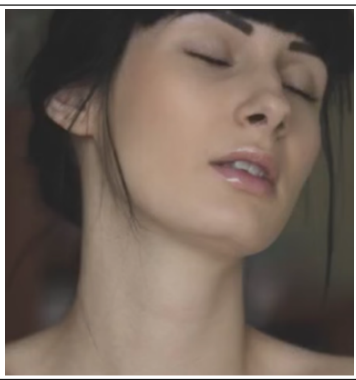
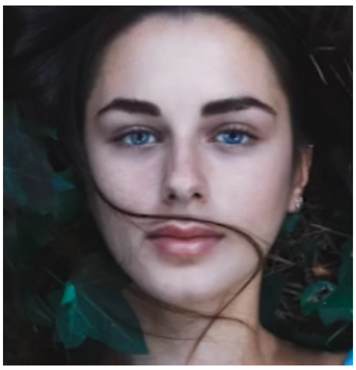
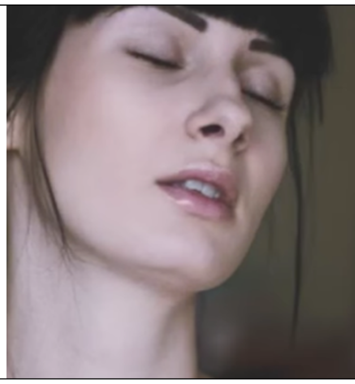
Phlearn also demonstrated a method for matching the skin colour of body and face in an image where the two appear mismatched [2]. The author sampled a range of colours from the body and adjusted the face with the levels tool for each colour channel. We show the results acheived in Table 3.

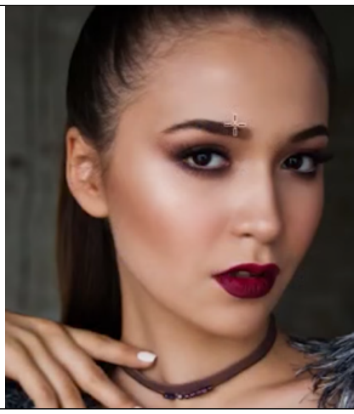
Table 3: Screen captures from Photoshop tutorial for matching the skintones of face and body.

Original	Target	Result
		

PiXimperfect demonstrated a method for matching skin colour in one portrait to another [3]. PiXimperfect first calculates the two average colours of the faces and uses the Photoshop curves tool to match the average colours of the original image to the target image. There must then be further adjustments by eye to change colour, brightness and contrast. Examples of the results from PiXimperfect is show in Table 4

Table 4: Screen captures from Photoshop tutorial for matching the skintones of portraits of different people.

No.	Original	Target	Result
4.1			

4.2			

In summary, for most of the techniques surveyed, levels and curves are frequently used for small brightness adjustments [1–3], and often to reduce the vividness of the colour adjustments the saturation must be slightly decreased [1, 2]. After all other effects are applied, the opacity of the overall effect is often reduced from 100% for a more natural appearance [1, 2].

Limitations of Photoshop techniques

Unlike the purpose of our project, the Photoshop techniques surveyed are not meant for automation. Instead, they are meant to be tailored to each specific image that a human is adjusting, and there are many junctures where the specific numerical amount of an adjustment often have to be judged by eye. While Photoshop has a method for automating processes using actions, the processes are meant for increasing ease of use by artists who can make additional adjustments and are familiar with the tool, rather than for use in commercial applications where the process is entirely automated [19].

Another limitation is that Photoshop operates at a higher level of abstraction than image processing software making use of libraries such as OpenCV. Image processing code has much more control over processes that can be applied to images, and the regions on the image that processes are applied to.

Finally, some Photoshop effects may be proprietary and are of course limited to the platforms that Photoshop supports, while a program developed with a platform such as OpenCV can be made open source and adapted to uses on a variety of different platforms.

2.2 Academic work related to colour transfer and skin colour enhancement

We have also surveyed the current state of the academic work relevant to our project, which fall into four rough categories:

Colour transfer for general images. There is a large body of work on the subject of automatically transferring the “style” or colours in an example image to another image.

Though the work in this area is usually focused on being effective for a wide range different images, skin colour transfer algorithms and processes often refers to this body of work and so this area is of interest to us.

Colour transfer for human skin colour. There have also been several prior studies transferring colour specifically for images wherein skin colour is prominent. These are most similar in purpose to our project we will discuss each study in detail.

Skin colour transfer as part of other applications We have also found several examples of practical application of skin transfer algorithms, where different application demonstrate practical uses of usually relatively simple skin transfer algorithm that is part of a larger project; we will discuss several of these projects.

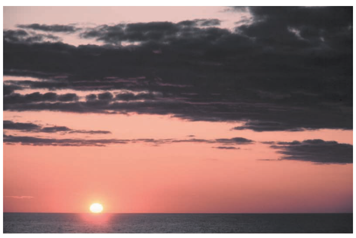
Skin colour enhancement mobile applications. Finally, there is the related field of skin tone enhancement software, where algorithms are usually intended to adjust the user skin colour towards a more pleasing tone and not to a specific target colour. We include the latter because unlike the other categories of prior work there are several studies of adjusting skin tone on a mobile device, which is part of the requirements for this project.

2.2.1 Colour transfer by example image for general images

Colour transfer refers to modifying the colours of an image to give it the desired appearance and style demonstrated by an example image, which we will refer to as the *target image*. Table 5 illustrates an example of this effect.

There have been a wide range of studies done in this area beginning with the seminal work of Reinhard et al. in 2001 [4]. The authors convert the image into $l\alpha\beta$ colour space, a colour space designed for natural scenes and based on research into human perception to reduce the correlation between each channel and remove the need to consider cross-channel effects when performing transformations on each channel. The authors then perform a simple operation to match the average and standard deviation of each channel of the original image to that of the target image. The resultant image is then converted back into rgb space.

Table 5: Example of image colour transfer using the algorithm from Reinhard et al. All images from [4]

No.	Original	Target	Result
5.1			

In a later study, Pitié et al. developed a method for entirely transferring the exact statistical distribution of the colours of the target image to the original image [12], and later improved on the technique with the motivation of automating film grading, or the process of enhancing frames in films to ensure consistency of colour and "feel" [5]. We show an example the effects they achieve in Table 6

Table 6: Example of film grading based on an example image using the algorithm from Pitié et al. All images from [5]

No.	Original	Target	Result
6.1			
6.2			

More recently, Bonnel et al. conducted a further study on colour transfer for film grading considering both spacial and temporal information [20] and Chang et al. created a tool for user editing of image based on a automatically generated colour palette [14].

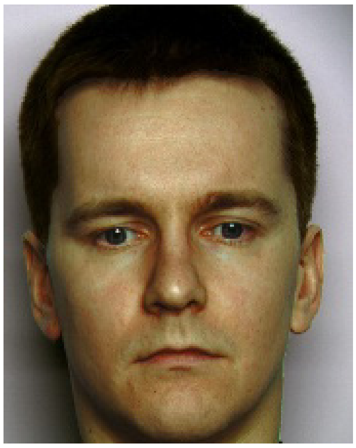
While these techniques are interesting possibilities to try when transferring human skin colour, because the these prior studies are all concerned with different problems that can arise with general images but not specifically for human skin colour, studies that specifically relate to human skin colour demonstrate that the general colour transfer techniques can be improved upon.

2.2.2 Colour transfer by example for images with human skin

There are fewer studies done specifically on the transfer of human skin colour, but there are several of great interest to us.

Seo et al. has a purpose most similar to the purpose of this project, which is to find a method of transferring human skin colours as realistically and accurately as possible [6]. In their study, authors demonstrate results that improve in realistic appearance compared to the Reinhard's algorithm. To acheive this, authors model the skin colour as an elongated distribution around a line referred to as the *principle line* in *rgb* space. To perform the colour transfer, the authors transform the distribution of the original image such that the principle line aligns with that of the target image. The authors then break the colour values into bins along the principle lines and also match the distribution of each bin.Table 7 demonstrates the output of this method.

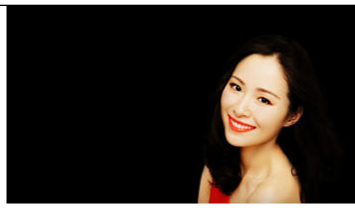
Table 7: Example of image colour transfer using the algorithm from Seo et al. All images from [6]

No.	Original	Target	Result
7.1			

However, as far as the other requirements of this project is concerned, it is not clear how fast this algorithm can run, particularly on a mobile device, nor the range of colours that the algorithm can transform a single skin colour, and it is in these areas that our project will attempt to improve upon.

Yang et al. performed a more recent study of colour transfer for human portraits [7] - Table 8 demonstrates their results. However, as Table 8 shows, this study focuses on the effect on the whole portrait, and places emphasis on transferring colour for different features of the human face. The actual algorithm the authors use to transfer skin colour is actually Reinhard's algorithm. This method also ranks the preferred target image for similarity to the original image before performing the colour transfer, which differs from our project where a key issue is that we have no control over the target image that the user will provide us and must be able to transfer to a wide range of colours.

Table 8: Example of image colour transfer using the algorithm from Yang et al. All images from [7]

No.	Original	Target	Result
8.1			

Yin et al. performed a study on the transfer of skin colour between races in order to aid a psychological study [8]. The authors use a process of first global adjustment of the two images to match the average colours, and then using the pixels from the target image with a colour most resembling the colour in the original image to entirely replace that pixel in the original image. Table 9 shows an example of the results using this technique.

Table 9: Example of image colour transfer using the algorithm from Yin et al. All images from [8]

No.	Original	Target	Result
9.1			

We include this study because it is one of the few that explicitly tries to transfer colour between large difference of skin colour, however, we feel it has many flaws with respect to our other requirements. Although the authors describe a study done evaluating the verisimilitude of the results with good results, we feel that particularly in the case of large skin colour changes shown above, the results are not realistic and show bright spots of saturation. Furthermore, the authors describe their algorithm as having $O(n^2)$ time, which likely means that this algorithm will perform too slowly for our purposes.

2.2.3 Skin colour transfer as part of other applications

Several applications performing different functions make use of skin colour transfer.

Shilkrot et al. published a study on transferring identity of the user on to a model image wearing garments the user may desire to purchase to create the virtual experience of the user trying on the garment [10]. The purpose of this article is fundamentally related to the purpose of the application that is this project's goal to support, and so this article is of great interest to us.

In fact, as part of the identity transfer, this article performs skin colour transfer on the model image to take on the skin colour of the user. Shilkrot uses a Gaussian Mixed Model for transfer skin colours and seems to use it for a relatively wide range of colour differences.

The difference between Shilkrot's study and ours is that skin is only a small part of their final image, and the skin transfer process is only a small part of their study, which also places emphasis on the transfer of the user's head and the reshaping of the model's body proportions. In our case, the hand will be the only object of interest in our inputs and outputs. While we need not devote our efforts to any aspect but the skin colour change, any flaws in the colour transfer causing unrealistic results will be much more noticeable.

Another interesting case of skin colour matching is the work Bitouk et al to create a face swapping software that seamlessly changes faces in photos to stock photo faces [21]. Since the skin colour of stock photo face does not exactly match the rest of the skin colour in the original photo, the authors adjust the lighting and skin colour until they do match. In their case however, the author specifically state large skin colour changes cannot be made, and to support a wide range of skin colours, the authors rely on having a large library of stock photos of every lighting position and skin colour. On the other hand, in our case, we are motivated by that fact that it is difficult to prepare videos for the full range of user skin colours.

Another application we've found is the work Baba et al. to develop a software that edits portraits in yearbook photos to all have a uniform skin colour given an example skin colour image [22]. The algorithm that the authors use to achieve this uses Piti's colour grading algorithm and guided image filtering. However, the goal of the project in terms of skin colour appears to be to have skin colours close to the target image but not necessarily exactly the same; the authors are focusing on the overall appearance of the set of portraits rather than the accuracy of the skin colour transfer for each individual image. On the other hand, the goal of our project is to ensure that the transformed skin colour of the model is as accurate as possible to the user's skin colour.

2.2.4 Skin colour enhancement mobile applications

For the most part the studies we have found are not meant to be run on a mobile platform and there are few specifically devoted to human skin colour. There are however many skin colour enhancement applications that modify human skin colour and several studies that perform this on mobile devices.

For example, Lee et al. enhances skin colour of users to a *preferred skin color*, or a skin colour close to the user's original skin colour that the user's race would find the most pleasing, in addition to changing the background of the user's video on a mobile phone [17]. They perform the skin colour transformation in HSV or hue, saturation and value colour space, simply adding the difference between the average hue and saturation values between the image and the target colour, damping the offset they add by a multiplicative factor for more realistic appearing results.

The difference between those studies and ours is that we have a target colour that could be very different from the colour that skin enhancement is aiming for, and our requirements for accuracy to the target colour is more stringent.

2.3 Summary of differences between prior studies and this project

In summary, we find that the prior work can satisfy some but all of our requirements for this project. Most of the sophisticated colour transfer work done is for general images and not meant to accurately and realistically transfer human skin colour. On the other hand, much of the work done related to correcting human skin colour is meant to improve the subjective appearance of human skin, and not meant to accurately change the human skin colour to a specific shade.

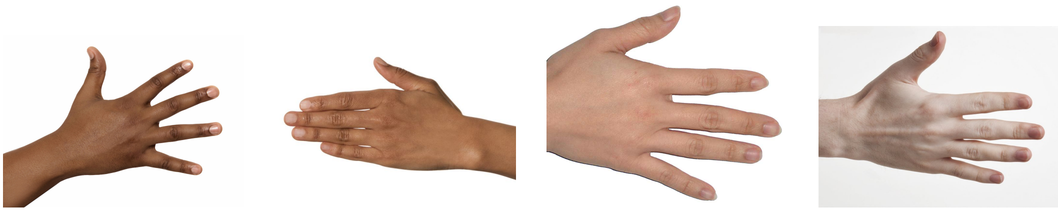
For studies specifically addressing the recolouring of human skin, many of the techniques are not meant to achieve skin colour transfer between a large difference in skin colour. The studies we found that did perform transforms across a large range often do not have sufficiently convincing and artifact-free results. Finally most of the results we surveyed do not perform the conversion at a speed fast enough for a mobile application, or else have not given the speed of the algorithm significant consideration.

3 Progress to Date

To accomplish the objective of recolouring the skin tone of a hand to a target colour, we wrote algorithms in C++ in Eclipse on OS X using OpenCV libraries. Eclipse is used to compile each iteration of the algorithm into a debug-mode executable program named Recolor. For ease of testing, as the algorithm is modified, we add more functionality to the Recolor program and retain the ability to use previous versions of the algorithm. We use a custom Python script to run new versions of Recolor from the terminal to test it. All of the relevant code and its versions are hosted on a git repository at https://github.com/tiantianhan/recolor

Recolor takes as input a hand image, a mask instructing it where to find the average skin colour of the hand, and a desired target skin colour. (Other flags and inputs are also used for testing purposes, see the Github repository readme file for a full description of the usage.) Recolor then outputs the processed image where the skin tone is adjusted to the target colour.

We iterated from simple to more complex algorithms, at each step testing the algorithm and evaluating the results. We tested progressive iterations on a set of hand images with varying skin tones. The images are are shown in Figure 1.



(a) (b) (c) (d)

Figure 1: Different hand images used for testing

For each test, we called the Recolor program to transform the image of one hand to have the skin tone of the hand in another image, then visually compared the processed image to the image of the target hand. We performed the process on all possible combinations of our test images, paying particular attention to the extreme cases, such as transforming from Figure 1a to Figure 1d and vice versa, as well as cases that start with a hand with mid-tone skin such as in Figure 1b (as this is the most likely use case for applications that change the image of a model's hand to match a range of skin tones). We evaluated the resulting images subjectively, based on whether the processed hand looks believably like a hand naturally of that skin tone, and noted any flaws that we then attempted to correct with the next iteration of the algorithm.

In the following subsections we summarize the results of each algorithm and our evaluation of the results.

3.1 Simple brightness addition / subtraction

Algorithm

To begin, we performed a simple addition of a value to each of the rgb channels of the hand, such that the average colour of the hand in the processed image is equal to the average colour of the hand in the target image. The algorithm is shown in Equation 1.

$$r' = r + \delta_r \tag{1}$$

Where

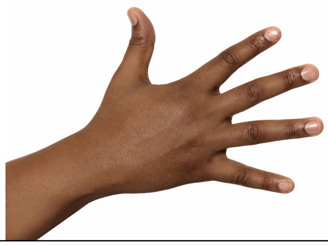
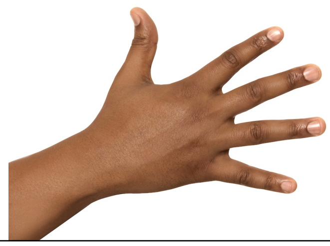
$$\delta_r = \bar{r_t} - \bar{r}$$

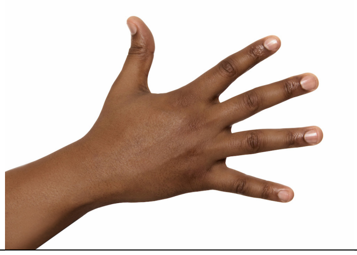
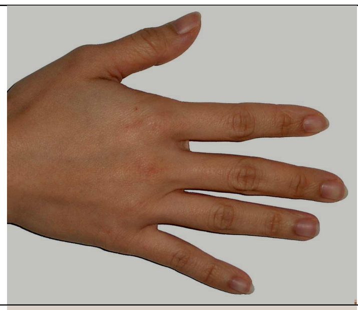
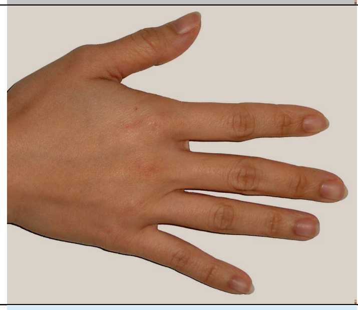
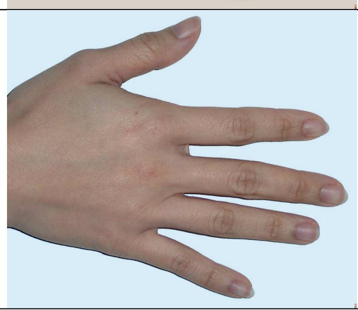
With the same equation applying for the g and b channels.

Results

The complete results are shown in Table 13 in Appendix A, and a portion is shown here for convenience.

Table 10: Test results of simple addition / subtraction brightening function.

No.	Original	Target	Results
10.1			
10.2			
10.3			

10.4			
10.5			
10.6			
10.7			
10.8			
10.9			

10.10			
10.11			
10.12			

Evaluation

Images of darker skin tones and smaller changes from the original skin tone to the target colour to begin with (Row 13.4) tend to have better results than images with large changes (Rows 13.2, 13.6). In the case of changes towards brighter colours, this is because large changes force bright points in the original image to be truncated at white, and also causes dark regions on the image, such as shadows and grooves, to become significantly brighter and less close to true black, giving the image a “high-key” appearance (Row 13.2 and 13.5).

In addition, we noted that at this stage the transformation from a dark coloured hand to a very pale hand, or even from a mid-toned hand to a pale hand and vice versa is especially unconvincing. (Row 13.6, also see 13.3 and 13.10)

3.2 Proportional adjustment relative to average color

Algorithm

To correct for the effect of the bright spots in the image being over bright and the high-key appearance resulting from all the shadows being brightened, we used an algorithm that maps the black and white points of the image to the same value, and adjusts the colours in between to match the target average colour. The algorithm is shown in Equation 2.

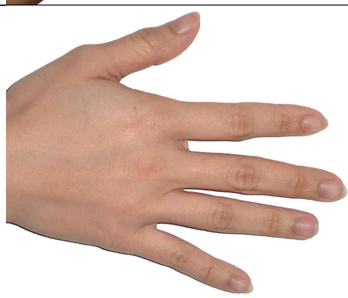
$$r' = \begin{cases} \left(\frac{\bar{r}_t}{\bar{r}}\right) r, & \text{for } r \leq \bar{r} \\ 255 - \left(\frac{255 - \bar{r}_t}{255 - \bar{r}}\right)(255 - r), & \text{for } r > \bar{r} \end{cases} \tag{2}$$

With the same equation applying for the g and b channels.

Results

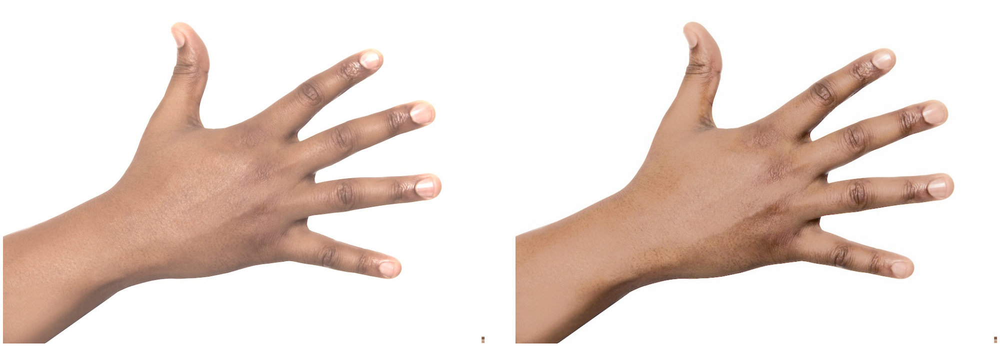
The complete results are shown in Table 14 in Appendix B, and a portion is shown here for convenience.

Portion of test results of adjusting proportionally based on distance of color to the average from Table 14 in the Appendix B

No.	Original	Target	Results
14.2			
14.4			
14.5			
14.6			

Evaluation

This method improved the appearance of cases with over-bright spots or “high-key” appearance issues, as Figure 2 shows:



(a) Simple addition algorithm (Equation 1)result

(b) Proportional adjustment algorithm (Equation 2) result

Figure 2: Comparison of algorithms 1 and 2 results for transforming a dark hand (Figure 1a) to a light hand (Figure 1c).

We noted however, that this method noticeably does not correct for, and even exacerbates slightly relative to the simple addition algorithm the dark spots at the joints and creases of a hand of darker skin tone when it is transformed to a lighter skin tone (Row 14.5). Other results are similar to the results of the simple addition algorithm.

3.3 Proportional adjustment with dark spot correction

Algorithm

We attempted to correct the dark spot issue by significantly reducing the absolute difference between dark pixels and the average colour, ensuring that the dark spots would instead have colours close to the average. We perform this correction on the output of the proportional adjustment algorithm.

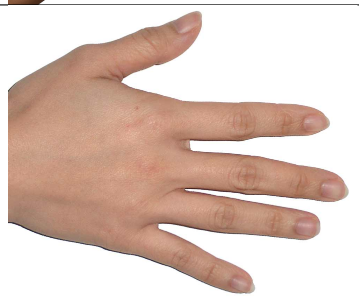
$$r'' = \begin{cases} \bar{r'} - \dfrac{(\bar{r'} - r')}{\alpha}, & \text{for } r' < \bar{r'} \\ r', & \text{for } r' \geq \bar{r'} \end{cases} \tag{3}$$

Where α is a constant, $\alpha > 1$. The same equation applies for the g and b channels.

Results

See Tables in Appendix C, for the complete results for $\alpha = 1.1$. A portion of the results is reproduced here for convenience.

Test results of proportional adjusting with correction for dark spots, $\alpha = 1.1$ from Table 15 in the Appendix C

No.	Original	Target	Results
15.1			
15.4			
15.5			
15.6			

Evaluation

As shown in Figure 3, the dark spots and creases noted in Section 3.2 are reduced.

(a) Proportional adjustment algorithm (Equation 2)result

(b) Proportional adjustment algorithm with correction (Equation 3) result

Figure 3: Comparison of algorithms 2 and 3 results for transforming a mid-toned hand (Figure 1b) to a light hand (Figure 1c).

We tried this effect for a range of α and found that $\alpha = 1.1$ gives an acceptably realistic result. A larger α would further reduce the dark spots on the skin but may begin to strongly brighten the shadows of the image, resulting in an unrealistic effect.

Up to the current iteration the more extreme changes of luminosity, such as from Figure 1a to Figure 1d and vice versa are especially unrealistic. Part of the reason is that the shadows, most prominent in Figure 1d may be causing the average colour of the entire hand to be of lower luminosity than it should be.

4 Future Work

References

[1] J. Shaver. (2014) How to change a person's skin color from dark to light in photoshop. [Online]. Available: http://www.designpanoply.com/blog/how-to-change-a-persons-skin-color-from-dark-to-light-in-photoshop

[2] Phlearn. (2014) How to match skin tones in photoshop. [Online]. Available: https://www.youtube.com/watch?v=5-DE49Muc0g

[3] PiXimperfect. (2017) How to match skin tone in photoshop cc 2017. [Online]. Available: https://www.youtube.com/watch?v=zEeknlHydfE

[4] E. Reinhard, M. Ashikhmin, B. Gooch, and P. Shirley, "Color transfer between images," *IEEE Computer Graphics Applications*, vol. 21, no. 5, pp. 34–41, Sep. 2001.

[5] F. Pitie, A. Kokaram, and R. Dahyot, "Automated colour grading using colour distribution transfer," *Computer Vision and Image Understanding*, vol. 107, no. 1-2, pp. 123 – 37, 2007/07/.

[6] K. C. Seo, G. Shin, and S. W. Lee, "Interactive transfer of human facial color," in *Advances in Multimedia Information Processing - PCM 2005: 6th Pacific Rim Conference on Multimedia, Jeju Island, Korea, November 13-16, 2005, Proceedings, Part II*, Y.-S. Ho and H.-J. Kim, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 993–1003.

[7] Y. Yang, H. Zhao, L. You, R. Tu, X. Wu, and X. Jin, "Semantic portrait color transfer with internet images," *Multimed. Tools Appl. (Germany)*, vol. 76, no. 1, pp. 523 – 41, 2017.

[8] L. Yin, J. Jia, and J. Morrissey, "Towards race-related face identification: Research on skin color transfer," in *Proceedings - Sixth IEEE International Conference on Automatic Face and Gesture Recognition*, Seoul, Korea, Republic of, 2004, pp. 362 – 368.

[9] T. Zhang, L. Cao, and W. Y. Wang, "The impact of virtual try-on image interaction technology on online shoppers' purchase decision," in *Proceedings of the 8th International Conference on E-Education, E-Business, E-Management and E-Learning*, ser. IC4E '17. New York, NY, USA: ACM, 2017, pp. 6–10.

[10] R. Shilkrot, D. Cohen-Or, A. Shamir, and L. Liu, "Garment personalization via identity transfer," *IEEE Computer Graphics and Applications*, vol. 33, no. 4, pp. 62–72, July 2013.

[11] L. Jun, Z. Mingmin, and P. Zhigeng, "Virtual try-on by replacing the person in image," *J. Comput. Aided Des. Comput. Graph. (China)*, vol. 27, no. 9, pp. 1694 – 700, Sept. 2015.

[12] F. Pitie, A. C. Kokaram, and R. Dahyot, "N-dimensional probability density function transfer and its application to colour transfer," in *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2, 2005, pp. 1434 – 1439.

[13] X. Chen, D. Zou, J. Li, X. Cao, Q. Zhao, and H. Zhang, "Sparse dictionary learning for edit propagation of high-resolution images," in *2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Los Alamitos, CA, USA, 2014, pp. 2854 – 61.

[14] H. Chang, O. Fried, Y. Liu, S. DiVerdi, and A. Finkelstein, "Palette-based photo recoloring," *ACM Trans. Graph.*, vol. 34, no. 4, pp. 139:1–139:11, Jul. 2015.

[15] Q. Zhang, C. Xiao, H. Sun, and F. Tang, "Palette-based image recoloring using color decomposition optimization," *IEEE Transactions on Image Processing*, vol. 26, no. 4, pp. 1952 – 1964, 2017.

[16] H. Aradhye, G. D. Toderici, and J. Yagnik, "Adaptive, selective, automatic tonal enhancement of faces," in *Proceedings of the 17th ACM International Conference on Multimedia*, ser. 2009. New York, NY, USA: ACM, 2009, pp. 677–680.

[17] J.-H. Lee, J. Heo, and Y.-S. Ho, "Skin tone enhancement and background change for mobile phones," *IEEE Transactions on Consumer Electronics*, vol. 56, no. 4, pp. 2719 – 2726, 2010.

[18] Phlearn. (2016) How to get a tan in photoshop. [Online]. Available: https://www.youtube.com/watch?v=Har8CSjei2k

[19] M. Woloszynowicz. (2014) How to easily correct colors and match tones in photoshop. [Online]. Available: http://www.vibrantshot.com/how-to-easily-correct-colors-and-match-tones-in-photoshop/

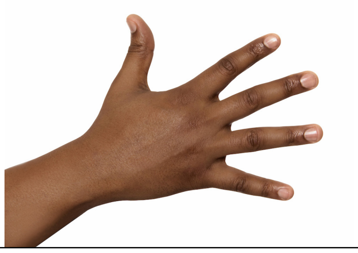
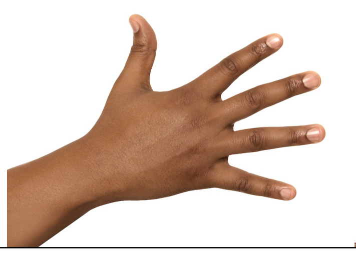
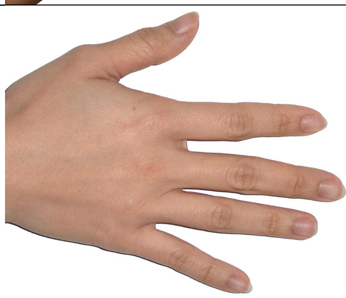
[20] N. Bonneel, K. Sunkavalli, S. Paris, and H. Pfister, "Example-based video color grading," *ACM Trans. Graph.*, vol. 32, no. 4, pp. 39:1–39:12, Jul. 2013.

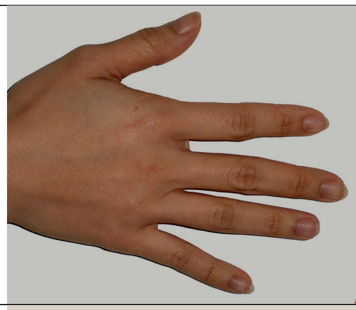
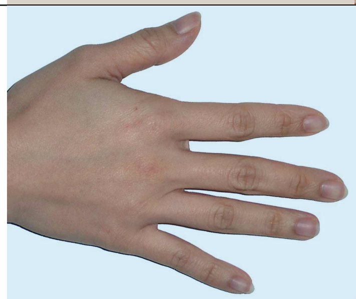
[21] D. Bitouk, N. Kumar, S. Dhillon, P. Belhumeur, and S. Nayar, "Face swapping: Automatically replacing faces in photographs," *ACM Transactions on Graphics*, vol. 27, no. 3, pp. 39 (8 pp.) –, 2008/08/.

[22] T. Baba, M. Okuda, P. Perrotin, T. Yusuke, and K. Shirai, "An automatic yearbook style photo generation method using color grading and guide image filtering based facial skin color correction," in *2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR). Proceedings*, Piscataway, NJ, USA, 2015//, pp. 504 – 8.

A Complete results for simple brightening

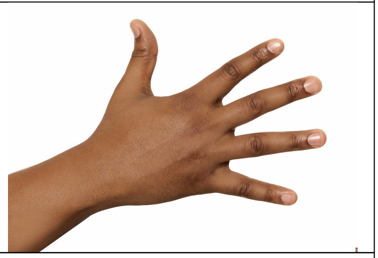
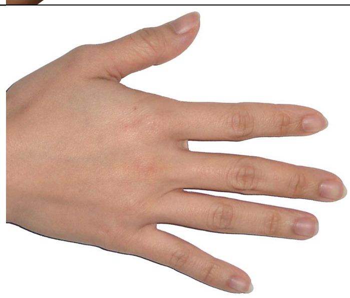
Table 13: Test results of simple addition / subtraction brightening function.

No.	Original	Target	Results
13.1			
13.2			
13.3			
13.4			
13.5			
13.6			

13.7			
13.8			
13.9			
13.10			
13.11			
13.12			

B Complete results for proportional brightness adjustment

Table 14: Test results of brightening proportionally based on distance of color to the average.

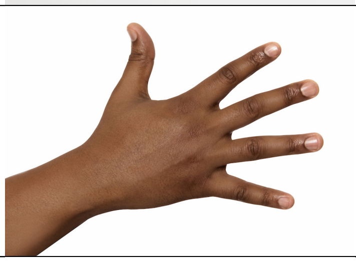
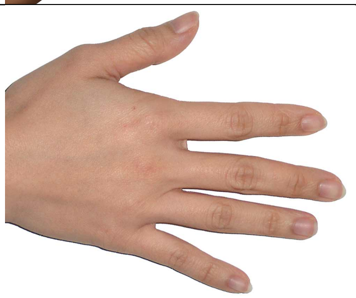
No.	Original	Target	Results
14.1			
14.2			
14.3			
14.4			
14.5			

14.6			
14.7			
14.8			
14.9			
14.10			
14.11			

14.12			

C Complete results for proportional adjustment with darkspot correction, $\alpha = 1.1$

Table 15: Test results of proportional brightening with correction for dark spots, alpha = 1.1

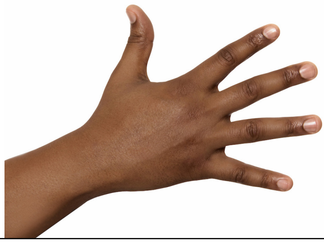
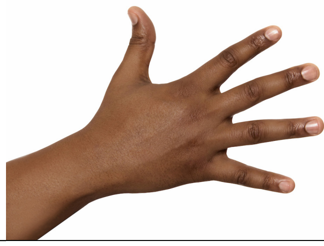
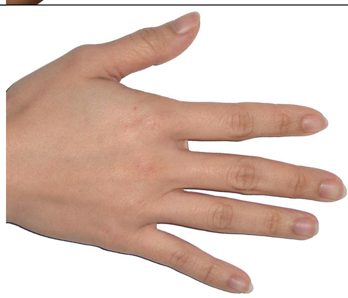
No.	Original	Target	Results
15.1			
15.2			
15.3			
15.4			
15.5			

15.6			
15.7			
15.8			
15.9			
15.10			
15.11			

15.12			

D Complete results for proportional adjustment with darkspot correction, $\alpha = 1.1$, calculating target average color with 5th percentile bright pixels

Table 16

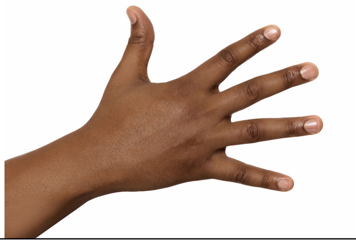
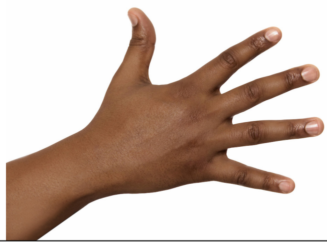
No.	Original	Target	Result
16.1			
16.2			
16.3			
16.4			
16.5			

16.6			
16.7			
16.8			
16.9			
16.10			
16.11			

16.12			

E Complete results for proportional adjustment with darkspot correction, $\alpha = 1.1$, calculating target average color with 10th percentile bright pixels

Table 17

No.	Original	Target	Result
17.1			
17.2			
17.3			
17.4			

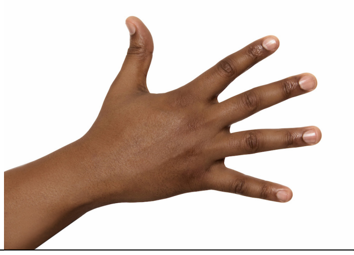
17.5			
17.6			
17.7			
17.8			
17.9			
17.10			

17.11			
17.12			

F Complete results for proportional adjustment with darkspot correction, $\alpha = 1.1$, calculating target average color with 25th percentile bright pixels

Table 18

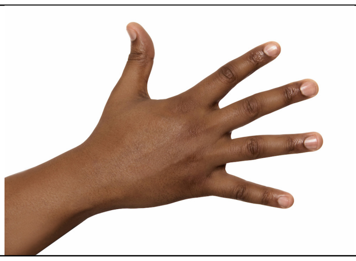
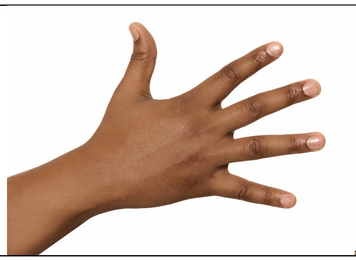
No.	Original	Target	Result
18.1			
18.2			
18.3			

18.4			
18.5			
18.6			
18.7			
18.8			
18.9			

18.10			
18.11			
18.12			

G Complete results for proportional adjustment with darkspot correction, $\alpha = 1.1$, calculating target average color with 100th percentile bright pixels

Table 19

No.	Original	Target	Result
19.1			
19.2			

19.3			
19.4			
19.5			
19.6			
19.7			
19.8			

19.9			
19.10			
19.11			
19.12