

Portrait Shadow Removal

CIS 581 Final Project

Grace Benner

gebennr@seas.upenn.edu

Chenchen Lin

linchenc@seas.upenn.edu

Isaac Wasserman

isaacrw@seas.upenn.edu

Abstract

Shadow removal is a computational photography task involving the detection and adjustment of shadowed pixels in an image to better match the remainder of the photo. Although supervised methods have historically been most successful at this task, advances in self-supervised image generation, such as GANs and more recently diffusion, have produced more data-efficient and generalizable methods for shadow removal. However, prior to the pervasiveness of neural networks for image processing, somewhat successful attempts at shadow removal were made using classical computer vision methods, such as thresholding, inpainting, and image decomposition. In this work, we narrow our scope to facial shadow removal and compare three promising methods, one supervised method, one based on self-supervised neural networks, and one novel algorithm based on classical computer vision techniques.

1. Introduction

1.1. Motivation

For our final project, we explored the computational photography task of portrait shadow removal. In an age where everyone is a photographer, we have come to expect the ability to take photos wherever and whenever. However, oftentimes, we are so focused on the subject that we neglect to consider their environment, specifically the ways in which it affects the lighting. While uniform color correction can easily fix underexposed and overexposed photos or attempt to fix mild exposure discrepancies in high contrast scenes, unevenly lit subjects are notoriously difficult to reconcile. Historically, harsh facial shadows could only be remedied by meticulous manual dodging and burning. However, more recently, computer vision techniques for recognizing and obfuscating harsh shadows have become increasingly prevalent, with many cameras applying these adjustments by default. Advances in generative machine learning methods have yielded new means for in-painting,

using large training sets as well as the surrounding image as context. However, such models are designed to be incredibly multipurpose and used for in-painting in various scenarios, which begs the question: is there a simpler, principled, and more data efficient solution to shadow removal?

1.2. Objectives

We seek to address this question by implementing and comparing three facial shadow removal techniques, the first being based on deep-image-priors [10], the second being Encoder–Decoder architecture with two-stage shadow modeling algorithm [12], and the third using classical computer vision techniques such as linear regression, Poisson editing, mixture models, and face-specific priors. We will compare the results of both methods on the portrait shadow manipulation dataset proposed by Zhang et al. (2020) [19], and we will discuss the merits of these methods over their counterparts as well as more multipurpose diffusion or vision transformer based techniques.

2. Related Work

Traditional shadow removal techniques in computer graphics typically follow one of several methods for identifying and eliminating shadows. These methods include manual labeling of shadow regions, leveraging shadow priors like illumination inconsistencies along shadow edges, and establishing connections between shadowed and non-shadowed areas. Subsequently, shadow elimination is achieved through histogram adjustment, color transformation, or through the use of illumination models and relighting techniques. The disadvantages of the traditional method is that you have to go through the process of identifying the shadows.

With the development of deep learning techniques, people have begun to employ neural networks for automatic shadow removal tools. DeshadowNet [13] adopts an end-to-end approach to shadow removal by predicting a shadow matte. ST-CGAN [16] utilizes a conditional GAN to generate both shadow-free images and shadow masks in a unified

framework. Meanwhile, Mask-ShadowGAN [11] and ARGAN [6] also harness generative models to perform shadow removal, either with unpaired training data or in a semi-supervised manner. These general-purpose methods cannot preserve the authenticity of the input faces. One reason is that these general-purpose networks are unable to capture specific face characteristics. For instance, human face skin is a highly scattering material that also has a complex absorption spectrum. BlindShadowRemoval [12] is a specialized supervised portrait shadow removal model that can better handle both subsurface scattering effects and color distortion. They also proposed a novel Temporal Sharing Module (TSM) to extract hierarchical features across multiple aligned video frames to adapt to portrait shadow removal in video scenarios, which greatly expands its applicability in diverse contexts.

However, most of these deep learning models are supervised, requiring both shadow-free portraits and shadow portraits. The acquisition of these paired images is tedious and may introduce some unwanted changes in lighting and other details. Also, this input pair means that we would have larger input size, causing longer training time. And in real life scenes, we normally only have the shadow portraits and want the model to be able to recover that. Thus, an unsupervised model is needed.

In 2021, He and Xing first proposed an unsupervised portrait shadow removal method via generative priors called ShadowGP [10]. While achieving comparable performance with supervised models, their proposed model is also more efficient and more generalizable.

Hand crafted shadow removal methods were a more popular area of research in the 2000s, with a number of analytical solutions to generalized shadow removal being developed using a combination of early inpainting techniques and image decomposition. Finlayson et al. (2002) [7] leverages an understanding of the physical phenomena of reflectance to design a mathematical model that decomposes an image into so called “intrinsic images,” a pair of images where one strictly corresponds to the chromaticity (hue and saturation) and the other strictly corresponds to the luminance component. Given accurate intrinsic images, shadow detection is trivial, requiring us to simply find edges with steep luminance gradients and negligible changes in chromaticity. Large swaths of uniformly illuminated (shadowed) pixels can subsequently be adjusted to match the surrounding illumination. In controlled settings, this technique worked quite well, especially in simple compositions and texture rich images which mask imperfections. However, this method relies on camera calibration to anticipate the effect of shadow on the image. Finlayson et al. (2005) [8] also attempts similar decomposition into these intrinsic images; however, it discards this assumption of calibration in favor of an image-specific optimization procedure. In this case,

a projection is found into a one-dimensional (purportedly chromaticity-only) space which minimizes entropy, forming one of the two intrinsic images, from which the second can be inferred. While this method is significantly more flexible than the previous iteration, it still often fails to produce convincing results when shadow boundaries cut through the most salient regions of the image. Considering our instinctive tendency to pay close attention to faces, this is especially unfortunate for facial shadow removal. Shor and Lischinski (2008) [14] manage to produce more accurately unshadowed colors by estimating and applying their affine color shifting model at each individual level of a Laplacian pyramid and flattening these layers into a final image. They claim that without this Laplacian multiresolution process, reconstructed pixels often appear low contrast and do not completely blend with their surroundings.

Each of the classical methods discussed above struggle with shadow boundaries. Although the illumination gradient is often very steep on shadow boundaries, the shift in illumination is not instantaneous, and applying a color adjustment under a binary mask will typically result in a visible seam where unchanged pixels meet edited pixels. Blurring these pixels and feathering the mask can mostly hide this boundary in highly textured or non-salient regions, but in smooth high-saliency regions, inpainting is necessary to produce convincing results. Despite this, Shor and Linischinski (2008) [14] is the only method discussed above which addresses this. They introduce a post-processing step which uses a relatively simple texture synthesis inpainting technique. This technique locates nearby regions to replace the inpainted pixels with and refines these replacements by locating more convincing seams using a graph-cut segmentation technique.

3. Methods

3.1. ShadowGP

ShadowGP was the first successful attempt at unsupervised shadow removal. Although many supervised methods had been proposed, ShadowGP aims to avoid the limitations that relying upon a finite set of training data to learn imposes. ShadowGP leverages the the deep, generative priors from the embeddings in the pretrained GAN model. These deep priors contain high-quality information on facial geometrics and appearances. To use these priors effectively for unknown shadow degredation, the creators of ShadowGP formulates the problem as a decomposition. A shadowed portrait can be broken down into a shadow free image, a full-shadow image, and a shadow mask. Once decomposed, the image is then reconstructed, and using progressive optimization, they iterate through each of 3 stages until the reconstructed image is as close to the original as possible. Then the shadow-free image is outputted.

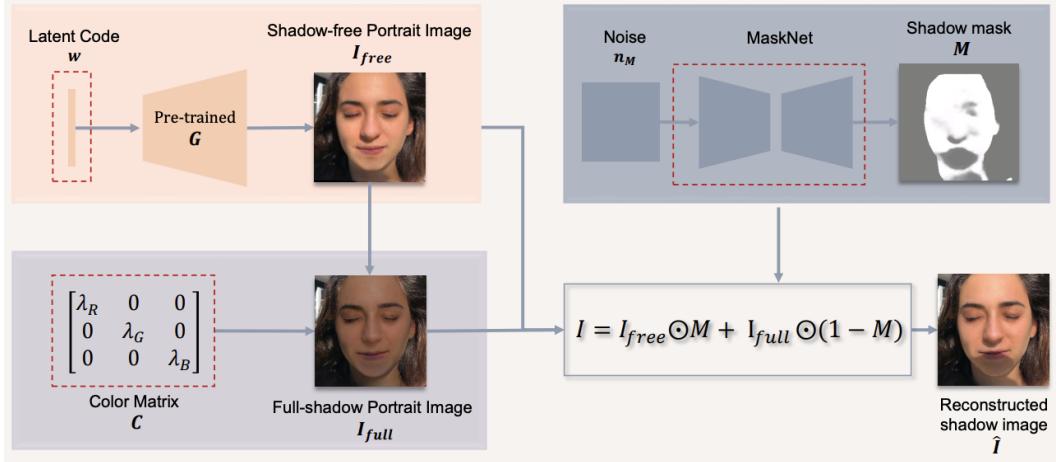


Figure 1. Overview of ShadowGP framework for decomposition.

More technically, given an input shadow portrait I their model aims to decompose it into

$$I = I_{free} \odot M + I_{full} \odot (1 - M) \quad (1)$$

where I_{free} is the shadow-free portrait image, I_{full} is the full-shadow portrait image, and M is the shadow mask (Fig. 1). The deep priors are used in construction of I_{free} , there is a learned color matrix C for the construction of I_{full} , and a MaskNet based on sampled noise to construct M . The GAN latent code w , color matrix C , and parameters for MaskNet all need to be optimized using their progressive optimization strategy.

The algorithm takes in a shadowed portrait I and a face parsing map S , and outputs the optimal I_{free} (as well as the I_{full} and M). Joint optimization produced sub-optimal results, so they divided the optimization process into three segments. Stage 1 is the initial face optimization. The model uses GAN inversion to project I into StyleGAN2 latent space to get an initial shadow-free face, using LPIPS loss for the initial shadow-free image I_{free}^{init} and the inputted image I as the optimization goal (see more on LPIPS in section 4), with a regularization term in the StyleGAN2 noise map. Stage 2 is the color matrix and shadow mask optimization, using the I_{free}^{init} from Stage 1. Within this stage, they are able to use joint optimization to acquire the color matrix C and parameters of MaskNet. This stage minimizes the $L2$ distance between the reconstructed shadow portrait

$$\hat{I} = I_{free}^{init} \odot M + (CI_{free}^{init}) \odot (1 - M) \quad (2)$$

and I . Stage 3 is the facial features refinement, again using the StyleGAN2 latent space, this stage aims to correct the reconstruction results in Stage 1 which may have missed some face details. This step minimizes a loss which is a combination of LPIPS for the facial features, and the LPIPS

between \hat{I} and I . Stage 3 updates both the color matrix C and the latent code w . Resulting in the final I_{free} , I_{full} , and M . All 3 stages use the ADAM algorithm to update what is being optimized in each iteration.

ShadowGP preformed better than some of the state-of-the-art portrait supervised shadow removal methods (in terms of LPIPS and SSIM), but not all. However, this method is not confined to any of the supervised method's generalization issues, due to its unsupervised nature. The authors demonstrated this by applying their method to portraits obstructed with watermarks, and even people with facial tattoos. ShadowGP was able to preform well in these new scenarios. One noted limitation of ShadowGP is its undesirable smoothing quality, where elements like wrinkles or busy beards may be altered. As well as some minimal lighting and color differences where the removed shadow was.

3.2. Blind Shadow Removal

In prior work, a further decomposition for shadowed images is updated from the equation (1). A gray-scale matte M_I represents the uneven intensity. In addition, the light outside the shadow region would penetrate beneath the skin, reach the vessels and reflect back, creating a red band around the shadow boundary. This subsurface scattering effect is modeled by M_{SS} , which is computed by blurring B with a different kernel per RGB channel. Now we have:

$$I = I_{free} \odot (1 - B \odot M_{SS}) + I_{full} \odot B \odot M_{SS} \odot M_I \quad (3)$$

More over, shadow region could be under light distortion, so we can formulate such color distortion by a 3×3 color transfer matrix C :

$$I_{full} = I_{free}C \quad (4)$$

What we are trying to get is a model $G(\cdot)$ to map from $I \rightarrow I_{free}$. The relationship between I and I_{free} in $G(\cdot)$ can be represented as:

$$W, N \leftarrow G(I | \omega), \hat{I}_{full} = I \odot W + N \quad (5)$$

However, if we continue to use equation (3)(4) from prior work, W and N become a function of I_{full} instead of being independent of I_{free} , due to presence of C matrix, which turns this problem into a memorization mode. So to deal with this issue, the Blind Shadow Removal method only looks at gray-scale image for first stage decomposition:

$$\hat{I}_{free,gs} = I_{gs} \odot (1 - B) + I_{gs} \odot B \odot M'_I \quad (6)$$

$$= I_{gs} \odot (1 - B + B \odot M'_I) \quad (7)$$

$$= I_{gs} \odot W \quad (8)$$

Now using this decomposition, $\hat{I}_{free,gs}$ can be directly expressed by the network weight. All that's left to do is colorization. As shown in figure 2, this model consists of two major steps: 1) grayscale shadow removal; 2) colorization. The model begins with a grayscale shadow removal module that predicts the deshadowed face in grayscale. The module consists of an encoder, a stack of residual non-local blocks, and a decoder. The encoder extracts features F from input images for shadow removal. It contains 4 convolution layers and 3 downsampling layers. To encourage spatial consistency for facial lighting and albedo, 3 stacked residual non-local blocks process the encoder features with positional encoding. The decoder then upsamples the features from non-local blocks via 3 transposed convolution layers, and estimates W and N . Here, positional encoding is from projected normalized coordinate code (PNCC) [20], which encodes the face semantics as vertexes. Colorization module breaks down into 3 steps: 1) erasing, 2) inpainting, and 3) color space transformation. Using the grayscale shadow removal module, we can get a shadow map:

$$\hat{B} = |\hat{I}_{free,gs} - I_{gs}| > \beta \quad (9)$$

The colorization module consists of 3 residual non-local blocks and a decoder. First, based on the shadow mask \hat{B} , we set the shadow region of F to be 0 to circumvent any potential disturbance, and denote it as the inpainting feature. Second, the inpainting feature $F \odot (1 - \hat{B})$ is concatenated with \hat{B} and the PNCC encoding, and fed to the module. The non-local blocks aim to fill in the missing region in F , and the decoder is designed to produce a M -channel color space $C \in \mathbb{R}^{N^2 \times M}$. In the end, we use three 1×1 convolution layers to transfer the grayscale face $\hat{I}_{free,gs}$ with the color space C back to the RGB face \hat{I}_{free} .

3.3. Novel Classical Technique

Images of faces constitute a relatively controlled scenario for image manipulation. Facial landmark estimators are widely available and have long been performant

and computationally inexpensive. Additionally, skin is often expected to appear smooth in portraits, allowing image manipulations to blur seams with minimal consequence. These characteristics make the problem particularly well-suited for handcrafted techniques which eschew complex neural modeling in favor of intentional and principled design choices.

To investigate whether such a model could compete with deep learning techniques, we built one of our own. Our model utilizes simple techniques to construct a pipeline which identifies shadowed skin regions, matches their color distribution to that of the unshadowed region, and hides editing boundaries.

We start by identifying facial landmarks (Fig. 3a), using a pretrained detector. The landmark detection models we found in our development process only captured landmarks from the eyebrows down. These features serve a number of purposes throughout the pipeline, but primarily, they allow us to generate a mask of the face (Fig 3b) so that we can ignore the torso, hair, and background in our calculations and adjustments.

However, as shadows often exceed these bounds, we realized that without softening this upper boundary, we would end up with a harsh transition from adjusted to unadjusted pixels. To remedy this, we devised a vertex-coloring technique. First, we push our highest vertices up slightly to account for the blending radius. Next, we assign each vertex a “color”, zero for the highest vertices and one for the rest. Then, we create a Delaunay triangulation of the landmarks, and we assign each pixel a color based on its barycentric coordinates with respect to its triangle.

With a more accurate mask of the face, we use the unmasked pixels to estimate the lighting distribution of the face, using a two-component Gaussian mixture model. We fit this model in LAB space in an attempt to emphasize the independent salience of luminance. By clustering the pixels of the face with this model, we separate the face into a shadowed and unshadowed region (the cluster with the lowest mean luminance is selected as the shadow region). This yields a somewhat noisy shadow mask (Fig. 3c) which is cleaned up using an opening operation, i.e. an erosion of the mask followed by a matching dilation.

Next, we must decide how to adjust the pixels in the shadowed region. While we could naively assume that the color distribution should match that of the well-lit region, this would neglect to consider natural asymmetries in lighting and variations in skin tone. Instead, we treat skin color as a function of the pixel's (x, y) coordinates. Using the well-lit region as training data, we use linear regression to impute the expected color of the remaining pixels. This results in a smooth color map of expected skin tones (Fig. 3d). We then find the mean value of the pixels in the color map which are under the shadow map, yielding our target

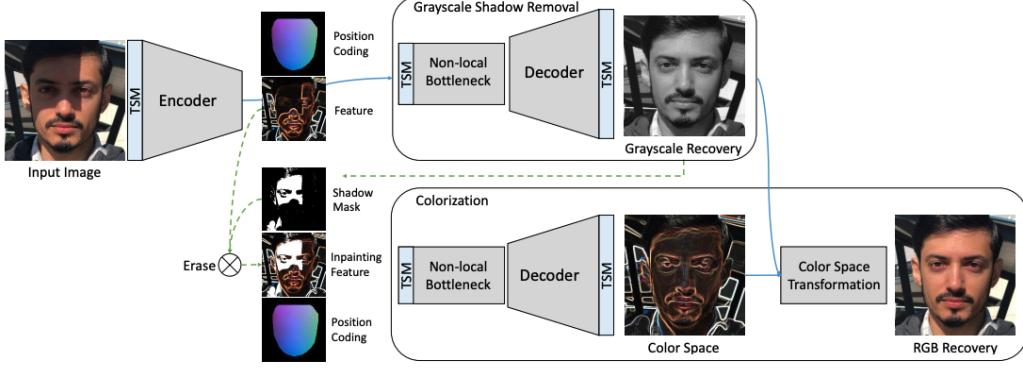


Figure 2. Blind Shadow Removal pipeline. (We did not use the TSM module because we only apply this model to image input.)

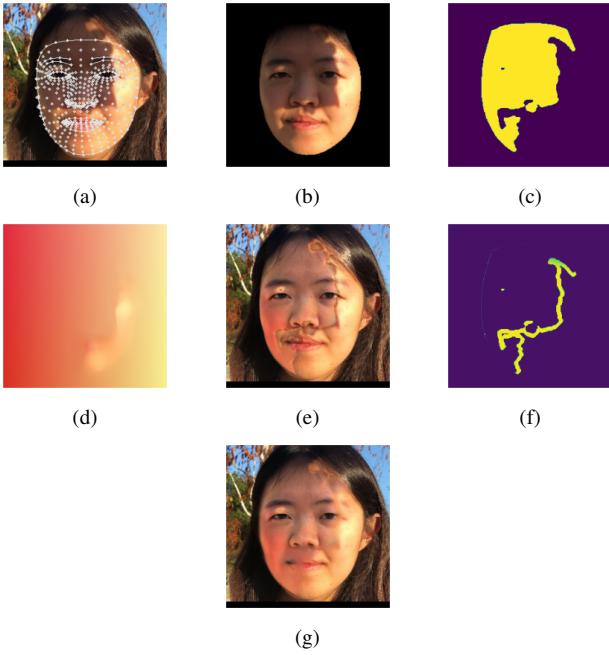


Figure 3. (a) Facial landmarks; (b) Masked face; (c) Estimated shadow mask; (d) Imputed color map; (e) Shadow adjusted image; (f) Boundary mask; (g) Final image with boundary inpainted

color. Pixels in the shadowed region of our image are each multiplicatively adjusted in order to match this target color.

As the transition into the shadowed region is not instantaneous, we are left with a shadowed boundary line which is neither well-lit nor adjusted by the previous step. Our attempts to feather our adjustment were unsuccessful, and as a result, we were left with the option of inpainting this region. Obtaining a mask of this boundary line by subtracting our shadow mask from a dilation of itself, we began by trying the inpainting algorithms built into OpenCV [4]. The first, from Telea (2004) [15] managed to soften the transition into the boundary region; however, it did very little to actually

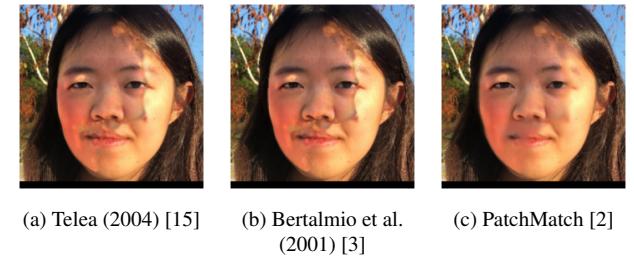


Figure 4. Comparison of the boundary inpainting methods tested

hide the discontinuity between the region and the rest of the face. The second, from Bertalmio et al. (2001) [3] behaved similarly, leaving the boundary region relatively unchanged. Having used modern image editing tools such as Photoshop [1], we knew that more accurate inpainting methods must exist to enable features like “content-aware fill” [9]. We found that this feature was enabled by a variation on a technique named PatchMatch [2]. This algorithm takes a relatively straightforward approach to inpainting, initially assigning each unknown image patch to a random known image patch and randomly reassigning the patch until a suitable match is found. These matches are propagated to nearby unknown patches to promote local consistency. As expected, PatchMatch inpainting greatly improved the fidelity of our image (Fig. 4c)

Inpainting can be performed on smooth areas with minimal artifacting. However, when the boundary area overlaps high-frequency elements such as the eyes or mouth, the blurring is distracting. We did our best to remedy this blurring by locating these salient regions using our landmark triangulation and applying a sharpening filter to these areas. This yields our final output image (Fig. 3g).

4. Results

4.1. Qualitative Comparison

We ran each of our methods on the same dataset, which contains 101 portraits that are all obstructed by shadows. All of these images were taken in natural lighting, and many of the subjects are repeated a handful of times. We will compare our results to a simple baseline which uses gamma adjustment (Fig. 5). This baseline reduced the appearance of shadows, by essentially brightening the whole image. It is one of the “quick-fixes” that will show up online when trying to remove shadows.

Beginning with ShadowGP, the results for Fig. 5i and Fig. 5j are visually pretty good shadow-free portraits of the original images. In Fig. 5i we do see some lingering shadow, however it is much lighter. Fig. 5j appears to us as the model’s best shadow-free result amongst the 4 outputted. The face shape, coloring, and his facial hair all look nearly identical to the original. ShadowGP does end up removing the shadow from his glasses, which the other methods do not do. In Fig. 5k and Fig. 5l ShadowGP does not perform quite as well. There are still some lingering shadows, but most importantly the eyes are affected. Interestingly, the eyes are affected in different ways, where in one they are much darker, and one much lighter. ShadowGP did mention that some limitations were a blurred effect and removal of other shadows or dark areas, so what we saw in Fig. 5j and Fig. 5l seem consistent with this idea. It is surprising however, that this was the method that also had trouble removing all the shadows.

Next, we examine the results of Blind Shadow Removal. Overall, we notice that the coloring of the image seems the least consistent with the original image. Blind Shadow Removal seems to have an issue with blending the shadowless area with the area the shadow was removed. We can see a border in all 4 figures. In Fig. 5m and Fig. 5n the area that the shadow was removed from tends to be darker than the shadow-free skin tone, while in Fig. 5o and Fig. 5p the area that the shadow was removed from is somewhat lighter than the shadow-free skin tone. Overall, the facial structures are pretty similar to the original image, with exception to Fig. 5o, which has some issues with the reconstruction of the nose and mouth.

Finally, looking at our novel classical approach, we see some slight discoloration near the borders of where the shadow was removed. The area of the shadow also tends to be slightly darker than the shadow-free skin tone. This method does result in a facial structure that is very close to the original. This method seemed to have the most trouble with Fig. 5q and Fig. 5t, in terms of blending the area where the shadow was removed. The novel classical did seem to remove some of the facial hair in Fig. 5r, however, this approach did leave the most realistic glasses shadow of all the

methods. Its performance on Fig. 5s is arguably the best of all the methods, since none of the facial features were distorted, and all three methods have some form of discoloration.

Overall, each method has its pitfalls. We think that visually, ShadowGP is the worst performing method for this set of images. It is highly inconsistent with its issues, which is the most worrisome. Using the deep priors from GAN might be hindering this method’s ability to match the original facial features, since these priors might be influencing the features. Both Blind Shadow Removal and Novel Classical have some discoloration issues, as well as obvious borders along the shadow line, but Blind Shadow Removal seems to have a less harsh distinction. With the exception of Fig. 5o vs. Fig. 5s, Blind Shadow Removal performs the best. It is important to note that in terms of computation time, the Novel Classical approach is the best. The other two methods rely on GPU use, whereas Novel Classical can be run on CPU’s and takes less than 0.5s on each image. A slight trade off in the quality, may be worth the time and energy saved running the Novel Classical versus the other deep learning methods.

4.2. Quantitative Comparison

4.2.1 Performance Metrics

We evaluate the models’ performance on the shadow removal dataset from [19] using two standard metrics.

The first is structural similarity (SSIM) [17], a patch-based metric which independently considers differences in luminance, contrast, and structure for corresponding image patches I_1 and I_2 . Luminance differences are quantified by

$$l(I_1, I_2) = \frac{2\mu_{I_1}\mu_{I_2}}{\mu_{I_1} + \mu_{I_2}} \quad (10)$$

where μ_{I_1} and μ_{I_2} are the mean pixel values of I_1 and I_2 , respectively. The contrast difference component is similarly defined as

$$c(I_1, I_2) = \frac{2\sigma_{I_1}\sigma_{I_2}}{\sigma_{I_1} + \sigma_{I_2}} \quad (11)$$

where σ_{I_1} and σ_{I_2} are the standard deviations of I_1 and I_2 . Finally, the structural component is given by

$$s(I_1, I_2) = \frac{\sigma_{I_1 I_2}}{\sigma_{I_1} \sigma_{I_2}} \quad (12)$$

where $\sigma_{I_1 I_2}$ is the covariance of the two patches. In practice, small constants are used in each component to avert division by zero. The structural similarity index between two patches is calculated as the product of these three components. This formula is applied to all corresponding patches in the images, and their indices are averaged to score a complete image.

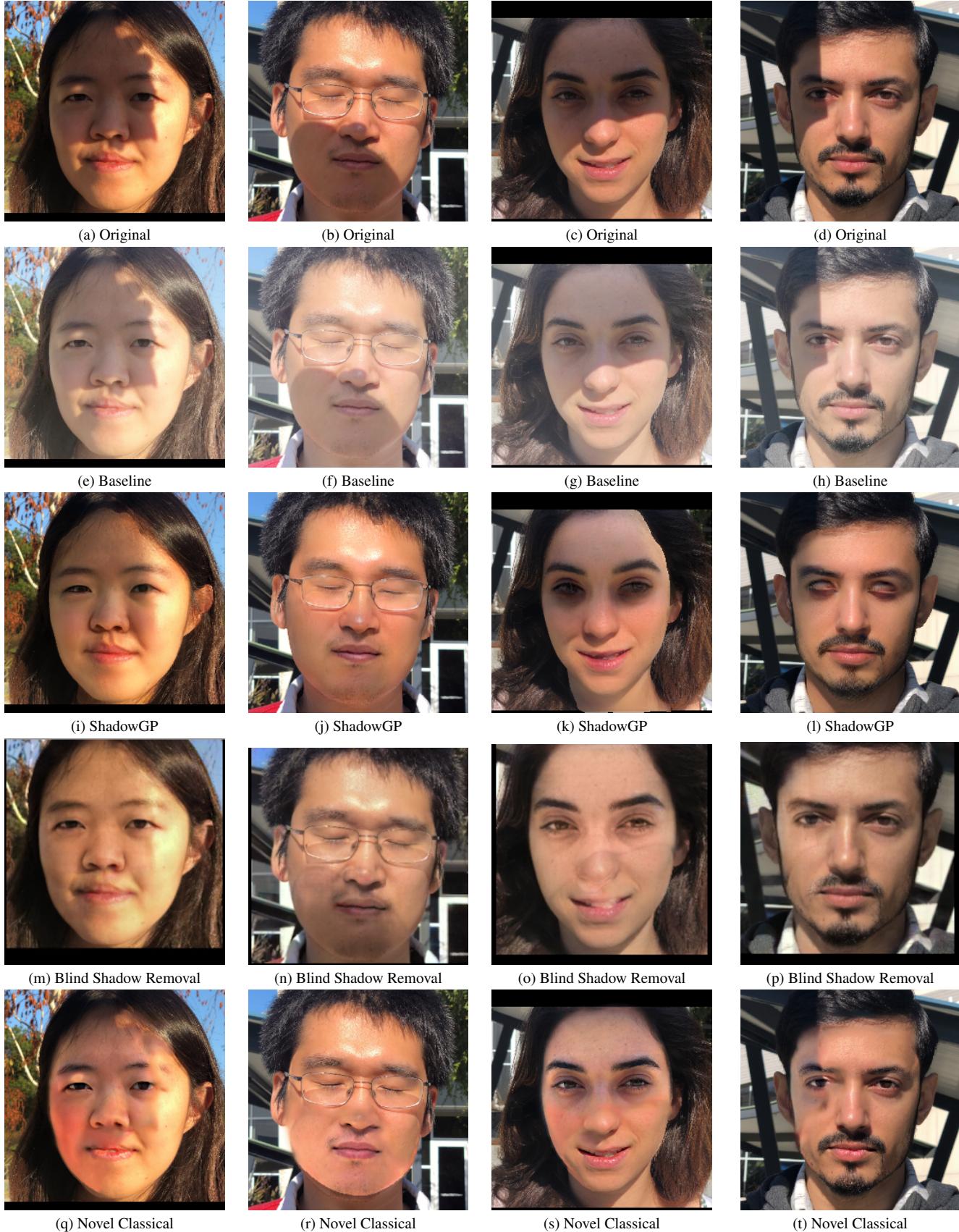


Figure 5. Comparison of the different shadow removal methods

The second metric, learned perceptual image patch similarity (LPIPS) [18] is a deep learning-based model of similarity which compares the deep features of the two images. Using a convolutional neural network, pretrained on ImageNet [5], LPIPS extracts localized deep features. The features are fed through an additional perceptual similarity model which produces a series of patch level difference indices which are averaged to give us the LPIPS loss.

4.2.2 Scores

Method	SSIM \uparrow	LPIPS \downarrow
Baseline	0.665	0.153
ShadowGP [10]	0.731	0.108
BSR [12]	0.783	0.135
Ours	<u>0.757</u>	0.117

Table 1. Comparison of mean structural similarity and learned perceptual image patch similarity loss (lower is better). Gamma adjustment is used as a baseline. Bold denotes the best score, while the underlined value is second best.

We were initially surprised by our quantitative analysis of the models’ performance (Table 1). In its original publication, ShadowGP [10] boasted SSIM indices of over 0.8, rivaling even its supervised counterparts, but in our tests, the model had significantly lower SSIM (lower than our hand-crafted model). However, their publication also cites slightly higher LPIPS than we found. Similarly, while [12] cites SSIM scores of over 0.86, we found scores that were nearly 10% lower. ShadowGP produced the lowest LPIPS indices in our tests, which is unsurprising, as the model uses this metric as a loss function, directly minimizing this quantity (perhaps at the expense of other metrics). Similarly, using a loss function derived from structural similarity, BSR most effectively maximizes this quantity.

Interestingly, however, our hand-crafted model ranks second in both of these metrics, despite never intentionally optimizing these quantities. Figures 6 and 7 reveal that while, on average, our model performs second best in the comparison, its performance is considerably less consistent than BSR in SSIM. It seems that by both metrics, BSR is more consistent, which is corroborated by our qualitative analysis.

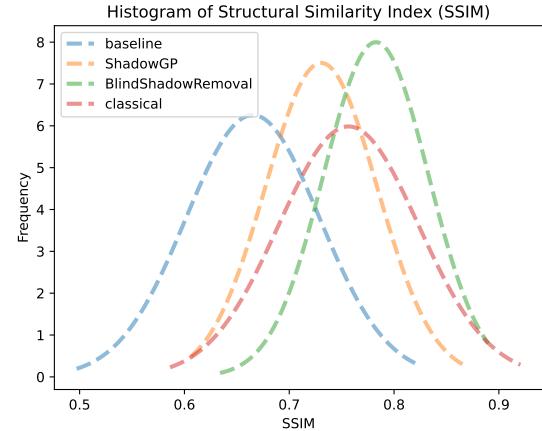


Figure 6. Histogram depicting the frequency with which each model produces output which is structurally similar to the ground truth.

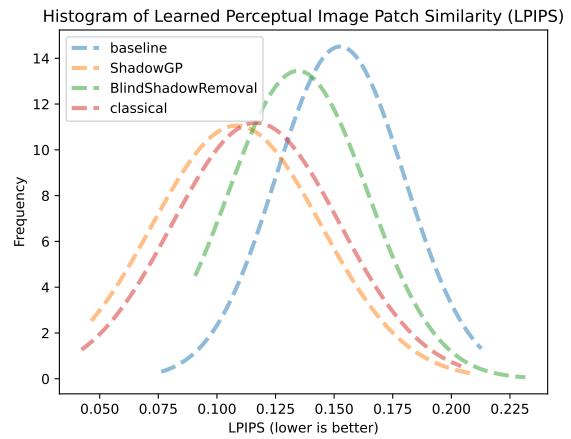


Figure 7. Histogram depicting the frequency with which each model produces output which is perceptually similar to the ground truth.

5. Future Considerations

After analyzing three different methods of portrait shadow removal, there are a few interesting considerations for exploration moving forward, beyond just improvements of the shadow removal. Firstly, after noticing the computation time to run many of these methods, it would be inefficient for a photographer to input all their portraits into a shadow removal method, if they didn’t all have shadows hindering the face. We propose a simple shadow classification model to be introduced to the pipeline to save time either spent sifting through all the images, or having a shadow-remover run on all images even if not needed.

We began to build a shadow identification classifier. This model will take in pairs of images and labels (1 for shadows, 0 for no shadows), and train a CNN to predict whether or not the portrait has a shadow. Since the data set we have been using was used to test pre-built models, it is not large or broad enough to create a model with great accuracy. There is also a lot of imbalance in the data, where shadow-free images are much less common. The model has trouble identifying these images, although did successfully predict some.

Another interesting area of portrait shadow removal to explore in the future would be extending these methods to videos. Running any of these methods on each frame would be computationally ineffective, so it would require building an optimizer to use information from the shadow in the previous frame, to limit the work needed to remove the shadow from the subsequent frame.

6. Conclusion

Overall, our experimentation with 3 different kinds of portrait shadow removal methods gave us valued insight. Our comparison of one self-supervised method, ShadowGP, and one supervised method, Blind Shadow Removal, to our own method using classic computer vision techniques, demonstrated that there is still a long way to go in automated portrait shadow removal. On visual inspection ShadowGP varied heavily in the quality of the output. While Blind Shadow Removal and our novel approach weren't perfect, many of the issues were consistent across the approach, such as discoloration or a shadow border. In numeric results, using both SSIM and LPIPS, we found nothing decisive either. Our novel approach ranked second in both metrics, with each of the deep learning methods taking first in one of these metrics. If we had to settle on the best performing method, we would choose Blind Shadow Removal. Our novel approach is however a close second, especially given the considerable efficiency over the others. After analyzing some of the pitfalls of both deep learning and classical portrait shadow removal methods, we ask ourselves if a subscription to Photoshop may be worth it after all?

References

- [1] Adobe Inc. Adobe photoshop.
- [2] Connelly Barnes, Eli Shechtman, Adam Finkelstein, and Dan B Goldman. PatchMatch: A randomized correspondence algorithm for structural image editing. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, 28(3), Aug. 2009.
- [3] Marcelo Bertalmio, Andrea L Bertozzi, and Guillermo Sapiro. Navier-stokes, fluid dynamics, and image and video inpainting. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, volume 1, pages I–I. IEEE, 2001.
- [4] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.
- [5] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.
- [6] Bin Ding, Chengjiang Long, Ling Zhang, and Chunxia Xiao. Argan: Attentive recurrent generative adversarial network for shadow detection and removal. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10213–10222, 2019.
- [7] Graham D Finlayson, Steven D Hordley, and Mark S Drew. Removing shadows from images. In *Computer Vision—ECCV 2002: 7th European Conference on Computer Vision Copenhagen, Denmark, May 28–31, 2002 Proceedings, Part IV 7*, pages 823–836. Springer, 2002.
- [8] Graham D Finlayson, Steven D Hordley, Cheng Lu, and Mark S Drew. On the removal of shadows from images. *IEEE transactions on pattern analysis and machine intelligence*, 28(1):59–68, 2005.
- [9] Dan Goldman, Eli Shechtman, Connelly Barnes, Ivan Bellunde, and Jeff Chien. Content aware fill. <https://web.archive.org/web/20170612085034/http://www.gartner.com/newsroom/id/1824919>, 2011. Accessed: 2020-04-29.
- [10] Yingqing He, Yazhou Xing, Tianjia Zhang, and Qifeng Chen. Unsupervised portrait shadow removal via generative priors. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 236–244, 2021.
- [11] Xiaowei Hu, Yitong Jiang, Chi-Wing Fu, and Pheng-Ann Heng. Mask-shadowgan: Learning to remove shadows from unpaired data. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2472–2481, 2019.
- [12] Yaojie Liu, Andrew Hou, Xinyu Huang, Liu Ren, and Xiaoming Liu. Blind removal of facial foreign shadows. 2022.
- [13] Liangqiong Qu, Jiandong Tian, Shengfeng He, Yandong Tang, and Rynson WH Lau. Deshadownet: A multi-context embedding deep network for shadow removal. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4067–4075, 2017.
- [14] Yael Shor and Dani Lischinski. The shadow meets the mask: Pyramid-based shadow removal. In *Computer Graphics Forum*, volume 27, pages 577–586. Wiley Online Library, 2008.
- [15] Alexandru Telea. An image inpainting technique based on the fast marching method. *Journal of graphics tools*, 9(1):23–34, 2004.
- [16] Jifeng Wang, Xiang Li, and Jian Yang. Stacked conditional generative adversarial networks for jointly learning shadow detection and shadow removal. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1788–1797, 2018.
- [17] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

- [18] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018.
- [19] Xuaner Zhang, Jonathan T Barron, Yun-Ta Tsai, Rohit Pandey, Xiuming Zhang, Ren Ng, and David E Jacobs. Portrait shadow manipulation. *ACM Transactions on Graphics (TOG)*, 39(4):78–1, 2020.
- [20] Xiangyu Zhu, Xiaoming Liu, Zhen Lei, and Stan Z. Li. Face alignment in full pose range: A 3d total solution. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(1):78–92, Jan. 2019.

A. Code and Results

Our results and code are available on Github at
github.com/isaacwasserman/FacialShadowRemoval.