Running Head: Automated Image Processing

Generating and Analyzing Adaptive Post-Processing Using Machine Learning

Jacob Mukobi

Camas High School

Abstract

As the world of technology moves towards end-user convenience and time-minimization, image editing must follow suit, specifically, color enhancement. The solution to minimize time and maximize convenience is to automate the process, of course. My approach to achieving this is to create a machine learning algorithm that uses hand-edited image data to predict how to adjust the colors on new images. In doing this, I found that color adjustment is much more nuanced than simply widening the dynamic range and emphasizing color. In reality, it is affected by things like color/brightness gradients, skin tones, and extremely bright and dark regions. In doing this, I found that the machine learning approach was very effective in improving the overall color and lightness signature of most images, but experienced drawbacks in the areas previously mentioned. With further development, this technology has the ability to allow anyone to adjust image colors as a professional would practically instantaneously, a much more advanced alternative to filters.

Generating and Analyzing Adaptive Post-Processing in Compact Photo Printers Using Experimental Trend Data

As society progresses into the future, technology is moving towards ultimate convenience on the part of the end user; with the advent of companies like Tesla and Waymo, even cars are driving themselves. In the arena of image editing, anyone can reach into their pocket and launch an app capable of applying one of a vast array of filters or tweaking a slew of image aspects by hand including contrast, saturation, sharpness, hue, and the list goes on. For the most detailed of us however, this can take what, compared to other modern conveniences, is lots of valuable time.

This is why I seek to create an autonomous, adaptive image editing algorithm that requires no end user decisions and takes virtually no time, and as society continues to progress, it's high time our image editors start driving themselves. Currently, there are a few developments in this field as well as a strong academic foundation of information to work from. This means that the success of this technology, though truly a modern innovation, will be on the shoulders of past intellectuals like Isaac Newton and Thomas Young.

Historical Overview

For much of human history, it has been widely assumed that the color white is light in its purest form and that the different colors are simply modified versions of that white light. It was not until Isaac Newton began experimentation with prisms that some light was shed on the topic. In his research, he found that when white light, when interrupted by a prism, is split into all visible wavelengths, but when the light of a single wavelength entered a prism, it was not able to be split again. Also, each different refracted wavelength was able to be recombined with others to create white light. From this, Newton gathered that white light was not simply the pure color of light, but that it was a "Heterogeneous mixture of differently refrangible Rays"(Mollon, 2003). This means that the prisms Newton was using didn't modify light to change its color, but rather separated it into its base components.

Moving from light to ink, the technology used to print in color can be traced back to a Dutch painter, Jacques Christophe Le Blon. Just after the turn of the 18th century, he began experimenting with mixing different paint colors. In the end, Le Blon became renowned for only needing three different base colors to be able to reproduce any color imaginable and his ability to accurately judge how much of each color to use, in a way, making Le Blon the first modern color printer. Also, like today's color printers, he even sometimes used a fourth color which was simply black in order to save ink and drying time. Le Blon began a company known as The Picture Office in 1721 whose goal was to mass-produce these trichromatic prints. Shares in the company soon rose up to 150%, but unfortunately, Le Blon was no businessman, and the company soon failed. However, he later published his work *Colorito* which laid out all the principles of trichromacy that Le Blon had discovered (Mollon, 2003).

This principle that Le Blon used is known as trichromacy, and three centuries later, it's this principle that light of different wavelengths can be combined in different proportions that makes up the basis of modern color science that is used in all electronic displays and printers. Specifically, Le Blon's method of composing color is known as subtractive color mixing, as opposed to Newtown who observed its additive counterpart. The difference between the two is that with subtractive, the paint or ink acts as a filter which absorbs or takes out (subtracts) certain wavelengths of light to produce the opposite colors. In this case, the more ink added, the darker the resultant image will become. Additive color mixing, as one might imagine, is quite the opposite. The actual light that will enter the observer's eyes is what is mixed. This results in a brighter image being produced with more light (Simonot and Hébert, 2013). In essence, more color in additive color mixing tends towards white, whereas more color in subtractive color mixing tends towards black.

At this point, the field had a time of stagnation as scientists debated various aspects of how color is produced and perceived. One example of this was whether light was made up of particles or waves. Newton claimed it was particles in his corpuscular theory which made it difficult to disagree with, as a result of his prowess in the scientific community, despite the various observations that hinted it might be a wave, such as the nature of light splitting at different angles through prisms, or the way it was both reflected and refracted in different mediums which had no explanation from the corpuscular theory. However, at the turn of the 19th century, a young physicist, physician, and, oddly enough, egyptologist named Thomas Young noticed these problems and offered the new hypothesis: light is a wave, not a particle, backed by stronger evidence. Young created the famous double-slit experiment to test his idea. In 1801, he presented his findings in the paper, *On the Theory of Light and Color,* to the Royal Society. The paper contained descriptions of the interference patterns of light and how he proved them with the double-slit experiment. Also, with his data, Young was able to measure the wavelengths associated with specific colors accurate to within a nanometer of modern measurments for some colors. He was able to relate the phenomenons of light to those of other waves that travel through air and water, proving Newton wrong (at least partially) saying “Much as I venerate the name of Newton, I am not therefore obliged to believe that he was infallible” in response to critics who believed Newton was (Tretkoff, 2008).

Thus cemented our modern understanding of additive and subtractive color reproduction and the principles that govern it. Since Thomas Young’s time and the development of the field of quantum mechanics, it has been realized that light behaves as a wave in some scenarios and a particle in others, not exclusively falling into either category. However, the theory associated with reproducing colors which most of us rely on each day in our printers and displays without even noticing can be traced back to these physicists who never could have guessed what their work would be used for in the future.

Current Trends and Practices

For the entire history of image capture and reproduction, people have tried to tweak the way their images to look, whether it's to make an image perfect for a museum or to attract more likes on popular social media platforms like Instagram. Originally, this process began in the darkroom, requiring hours of painstaking work of an expert with specialized equipment. As technology has advanced through the centuries and especially recent years, this process has become more and more accessible and convenient until today, at which point practically anyone can reach into their pocket and retrieve a phone with powerful apps that can alter images with stunning quality with the adjustment of a whole slew of sliders. As time and technology progress, the process will continue to be more convenient—but how? It may seem that simply choosing which buttons to press and where to position a few sliders may be as simple and easy as it could get, but those who believe this fail to account for automation: A concept which would mean editing images would become as convenient as selecting the photo, then letting a computer do the rest.

Without explicit automation, the nearest thing is image filters. As Saedeh Bakhshi states, "The goal of filters is to give photos a better exposure or stylized look without knowledge of photo processing" (2015). Though this can work fairly well for those who aren't quite sure what adjusting aspects like luminosity or chroma really does, it's always limited to a set number of unchanging filters that don't fit every image perfectly. However, problems arise when more are added because this makes the selection of the best one for a particular image a difficult process in and of itself. So, users who aren't comfortable manually altering images are stuck in a middle ground of a moderate amount of mediocre filters. Also, for some, this doesn't work at all, saying that "A lot of these apps, they just pile stuff on top of stuff on top of stuff" meaning that they just layer effects on top of the original image, rather than enhance it (Bakhshi, 2015). If a program existed which could automatically and adaptively edit photos, this would change the game, as it would have the potential to do what a more serious image editor could do with the simple push of a button that anyone could do.

A basic form of automated image editing is known as histogram extension. This method consists of reading pixel value data, then remapping each value from its relative position within the image to new values which are spread onto a much wider histogram to create a higher dynamic range (Pizer, 1987). While this does have certain limitations, mainly speed and quality, it is a reasonably effective way of enhancing images. One concern with this is that it can produce results that vary in effectiveness across the image. For example, if there is already a very bright region in an image but not much darkness, increasing contrast may make the darker regions look better but also blow out the bright parts. A solution to this is to evaluate the image data at the local level, so to adjust individual portions of images a more specialized and adequate amount (Stark, 2000). In the previous example, this would still help to enhance the dark sections, but not make the brightest parts too bright.

This type of automatic image enhancement software is being developed for consumer use by a group of MIT and Google artificial intelligence researchers with the goal of automatically editing images in the cloud so to emulate the work of a professional photographer. This software uses a form of machine learning to produce high dynamic range images. How the team achieved this was by using a large set of test images which were each edited by five different photographers. These unedited and edited images were then fed into a machine-learning algorithm which was able to make a connection between image data and how the photographers chose to edit the images. From that point, the resultant algorithm could take a new image and predict how the photographers would have chosen to edit the images.

A more advanced version of this is a software known as Luminar made by Skylum. Though not much information is publicly known on the specific approach, it does use similar machine learning algorithms to Google in order to automatically adjust various aspects of images, including contrast, saturation, highlights, shadows, and more. The goal of this is to expedite the image editing process into a single click so users have to spend none of the time to get resultant images that are as well edited as if they were done by a professional photographer. This software is much more powerful than what Google is doing currently, but is not as accessible, given it costs 93 dollars for a subscription to the service (Lusina, 2019).

As time continues, we should see the trend continue of the increased convenience of image editing. It started with expensive experts, specialized tools, and hours of time. Then, it progressed to expensive, powerful digital tools like photoshop which still requires a certain amount of expertise. After that, powerful apps became available on smartphones which could perform similar functions. However, this path will ultimately conclude when there is literally nothing that has to be done on the part of the user: fully automated image processing.

Limitations

Histogram equalization can be a very powerful technique to automatically edit images based on the histogram data. It has the ability to greatly improve contrast by extending the dynamic range to fill up more of the pixel value spectrum from pure white to pure black. As with anything, however, this technique does have its drawbacks. For images which have significant differences in their contrast characteristics in various regions of the same image, histogram equalization techniques that operate globally (gather data from the whole image, then change the entire image in the same manner) have the tendency to change the image too much or too little in those different regions (Stark, 2000).

A solution to this approach is to break images down into parts and perform local histogram equalization which compares each pixel to those around it, then changes each pixel differently compared to its local image characteristics (Stark, 2000). Even this method is not without its problems: Primarily speed. This solution essentially does the same thing, but many different times on each image. This means it takes more computing power, and in turn, more time.

Most importantly, whenever a raw image is altered in any way, the original data is also changed or lost. This means that any form of image editing results in a loss of detail, though sometimes very slight. In most images that are well-edited, this loss can be kept to a minimum, and details that already existed may even become clearer to an observer. However, in extreme cases, images can become incredibly distorted to the point of recognizability. But even with slight changes, " no matter what you do, edits degrade the data in an image file in three different ways: clipping, and tonal range expansion and contraction" (Schewe, 2011). For example, when contrast is raised in an image, making darks darker and lights lighter, pure white and pure black pixels cannot be made lighter and darker which means that pixels that are nearly pure white or black will become so and blend with the already pure white and black pixels (known as clipping). In this case, the location of where the pixels were different colors and the difference in luminosity is information that is lost. Also, if the midtones of an image were made brighter, this would decrease the variation between those midtones and the brightest parts of the image which *contracts* the lighter tonal range (tonal range contraction). Also in that same instance, the variation between the midtones and darkest parts of the image would increase. While this doesn't explicitly result in any data loss, color gradients are then stretched over a greater range which can challenge the illusion of a smooth gradient from dark to light (tonal range expansion) (Schewe, 2011).

In essence, when only considering the information contained in an image, it is impossible to add detail that existed in the original subject captured, but it is almost inevitable to lose detail when images are altered, which is why when editing images, it is of paramount importance to strike a balance between maintaining as much of the original information in the image and improving colors, dynamic range, and more. Now, in most images, these losses can go unnoticed, but only if the changes made are made in moderation.

Conclusion

Since Newton began to consider the nature of light and color reproduction, color science and the production of images has been advancing with rapidity. With this comes the desire to make these images look better, as well as a modern desire for convenience. The concept of automated image editing requires all these aspects to be successful which is why now is the time for it to arise.

To do so, a likely approach is to start using a machine-learned method of histogram equalization. This can then be developed into a local version of a similar technology that could avoid problems which arise with images that have varying brightness characteristics across the image. Eventually, this could be expanded upon with a facial recognition feature which determines if there are faces in the image, then would have the ability to smooth out skin, improve skin tones, and possibly even enhance features, such as enlarging eyes. Overall, this is a pivotal moment in technology as mere end products are reaching peaks, but the quality and ease of the user experience is changing more and more rapidly. Image processing is just one avenue of this trend, but an exciting one because, in time, anyone will have the ability to obtain the quality of editing that could be expected of a professional photographer at the touch of a button.

Materials and Methods

In order to understand how a machine learning algorithm can adjust image colors, we must first understand how one works. In my particular model, I used three single-neuron neural networks, one to adjust contrast, brightness, and saturation independently from one another. In each neural network, I used two input variables that were gathered by reading images themselves, then related them to how much I adjusted images by hand. Each neural network takes the two inputs, weighs them (multiplies them by a constant known as the weight), then inputs the sum of the weighted variables into a squashing function (takes inputs and converts them to reasonable end values.

In my particular setup, I created two separate sets of 100 images each, one being my training set, and the other, a testing set. With all the images, I adjust the contrast, saturation, and brightness by hand using the PIL library in python (more on this later). As I did this, I recorded the factors by which I changed these image aspects. As well as measuring hand adjustments, I used the same PIL library to convert the images into the HSV colorspace. This represents the data for each pixel as a 3-variable tuple including hue (what color it is), saturation (how intense the color is), and value (how light or dark the pixel is). In doing this, I could extract statistical data from images that could be related to how much I altered them by hand.

Specifically for contrast, I recorded the range of the value channel (the difference between the lightest and darkest pixels) as well as the standard deviation of that channel. For saturation, the standard deviation and median of the saturation channel were recorded. Finally, for the brightness, I recorded the standard deviation and median of the value channel.

In the machine learning algorithm, each neural net starts by weighting both variables at 1 (it inputs them directly into the squashing function). This squashing function then relates each input to a corresponding image adjustment coefficient (what I changed by hand and recorded). At this point, the algorithm measures the average error (the difference between what the squashing function outputted and what the hand-edited values were). It also performs this operation 4 other times, each separately with adjusting each weight + or - 1. Now, having the error for 5 different weight settings, the algorithm finds which has the least error. Then, it sets the weights of that original iteration as the new starting weights and begins the process again (starts a new iteration). When the weight setting that has the least error is the same as the previous iteration, the amount by which the weights are altered is lowered to 999/1000 of its previous amount. This allows the algorithm to "zero in" on the best possible weights by slowly increasing the precision of iteration. This process was then repeated with each, contrast, saturation, and brightness with 20,000 iterations each (at this point, the differences between iterations became insignificant, so not to justify doing more. At the end of this process, the final weights for each variable and the squashing function were recorded for all three image aspects.

Finally, a new program could be written. This, using the PIL library, can read images, extract the pixel data from any image (which is the derivation of the algorithm inputs). With these values, this program then multiplies each input by its respective weight, sums them, inputs them into the squashing function, and produces a final adjustment coefficient for the three image aspects that are intended to be accurate approximations of how they would be edited by hand. These coefficients can be used, similarly to when the images were edited by hand, automatically produce a new, altered image.

Results

To gain experimental result data, the testing set of images were used. In order to ensure that the algorithm doesn't only work for the relatively few images it was trained on, the testing set has to be completely different which means it was a group of 100 completely different images that the machine learning algorithm had never interacted with before. They were run through the final program that alters the images while recording the adjustment coefficients. These could be compared to the hand-adjustments to determine the accuracy of the algorithm.

Upon consolidating the results from this machine learning algorithm, the results were mostly positive. The average error which is defined by the difference between the automatically produced enhancement coefficient and hand coefficients was 8.17%, and the median error was 5.55%. This suggests that the algorithm is fairly accurate across the test set and that it has little skew, meaning it doesn’t tend to only either over or under-adjust images.

From a qualitative standpoint, the vast majority of images, testing and training look, subjectively, better than the original versions. This manifested itself in more emphasized colors, higher dynamic range, and balanced brightness. On the other hand, the system was not perfect: It primarily had issues with skin tones, mainly on faces. When raising contrast and saturation, the values for these have a point at which they cannot be raised any further. If the adjustments ask for significantly more than that ceiling, the differences in pixel data, especially in gradients, will be forced to compress and become less noticeable. This results in gradients developing a more uniform and less varied look. Also, on the other end of the spectrum (again mainly in gradients), differences can be forced to spread out across a larger range than the original data. This results in banding, which is the development of distinct bands of colors that do not transition smoothly into each other. Faces and other skin tones are very common examples of these gradients which, when experiencing these phenomena, tend to look very obviously unnatural to the eye.

Discussion

Overall, this machine learning algorithm can be considered a success, though it has room for improvement. The end-goal for this technology is to automate the enhancement of images (make them look better). For most images, it does this. Also, the error between hand adjusted images and the automatically adjusted is fairly low, meaning it is fairly accurate to my personal adjustment preferences.

However, the current version of the algorithm and program do have weaknesses. In machine learning, data sets are incredibly crucial detail. Most full-scale algorithms use massive data sets with thousands of data. As an individual, adjusting each image by hand was very time and effort-intensive which limited the training and testing sets to 100 images each, a relatively small number in the context of the algorithm. This, of course, presents the same issues that come with small sample sizes. For example, it is possible that the data sets are not accurately representative of all images that the algorithm would edit in practice. However, in its current state, the algorithm is quite accurate across the vast majority of images it sees, suggesting that this is not the case.

Also, the entire process of editing and analyzing all the images was done in an office building on one monitor with one person. The first problem that this presents is the fact that the environment the images were viewed in is definitely not representative of every environment in which it would be used once implemented. Also, though I have an in-depth understanding of color science, I am not a professional photographer. This means that, though I did my best to adjust the images to my preferences, they were likely not what most people may prefer. This means that the algorithm is tuned specifically to my preferences.

Before this technology would be implemented on a larger scale, it would be best to use multiple professional photographers and a much larger training set of images. Also, with regards to environmental effects on my perception of the images, I did my best to keep everything as accurate as possible, given the circumstances, throughout the entire process. To start, I calibrated the monitor which I was viewing the images on. I also attempted to control as many variables as possible: I was the only person viewing the images, I used the same monitor with the same settings, and I did so in the same office space with the same lighting.

Conclusion

So even though the current version of the project has its limitations, the approach of using a machine learning algorithm to automatically edit images shows a lot of potential, given that the small-scale version that can be considered a proof of concept was quite successful. With future development and a larger scale, the project will gain accuracy, adaptability, and reliability, becoming a significant contender with things like image editing apps and filters. This technology, though in its infancy may well be the next step in automating the future.

References

Bakhshi, S., Shamma, D.A., Kennedy, L., & Gilbert, E. (2015). Why We Filter Our Photos and How It Impacts Engagement. ICWSM.

Hardesty, L. (2017, August 01). Automatic image retouching on your phone. Retrieved July 25, 2019, from http://news.mit.edu/2017/automatic-image-retouching-phone-0802

Lusina, A. (2019, July 10). Luminar Flex Sees a Major Update and Introduces Accent AI 2.0 Filter. Retrieved from https://fstoppers.com/news/luminar-flex-sees-major-update-and-introduces-accent-ai-20-filter-388434

Mollon, J. D. (2003). The Origins of Modern Color Science. The Science of Color, 1-39. doi:10.1016/b978-044451251-2/50002-7

Pizer, S. M. (1987). Adaptive histogram equalization and its variations. Computer Vision, Graphics, and Image Processing, 38(1), 99. doi:10.1016/s0734-189x(87)80156-1

Schewe, J., & Fraser, B. (2011). Real world Camera Raw with Adobe Photoshop CS5: Industrial-strength production techniques. Berkeley, CA: Peachpit.

Simonot, L., & Hébert, M. (2013). Between additive and subtractive color mixings: Intermediate mixing models. *Journal of the Optical Society of America A,31*(1), 58. doi:10.1364/josaa.31.000058

Stark, J. A. (2000). Adaptive image contrast enhancement using generalizations of histogram equalization. IEEE Transactions on Image Processing, 9(5), 889-896. doi:10.1109/83.841534

Tretkoff, E. (2008, May). This Month in Physics History: May 1801: Thomas Young and the Nature of Light. Retrieved July 18, 2019, from https://www.aps.org/publications/apsnews/200805/physicshistory.cfm