4TN4 Project 2

Stephen Fu, George Zhang

April 2018

1 Introduction

In photography, it is sometimes the case that an image is taken under low lighting conditions and we wish to improve the visual quality to make out more details or colors in the image. These are typically referred to as *low contrast images*, images where lots of tones are saturated around a certain range of colors (these colors need not be dark colors).

In a visual sense, contrast is the difference in luminance or color that makes different parts of an image distinguishable (Wikipedia). A high contrast implies that an object in the image should have a larger difference in pixel values with nearby objects or background. In some sense, we can use this color difference to measure contrast.

A lot of algorithms have been created to try to address this issue. In this project we explore some old techniques such as histogram equalization, contrast stretching briefly before proceeding to implement the main technique, optimal contrast tone mapping (OCTM).

2 Previous Methods

This section is a brief discussion on previous methods. Results and comparisons of each method will be provided in the section Experimental Results.

2.1 Histogram Equalization (HE)

Histogram equalization, sometimes used as a contrast enhancement method, will shift the bins of image histogram over the entire domain in such a way that the cumulative density function is as close as possible linear over the domain. Even though this does have the effect of spreading out the image tones, this does not necessarily correlate to desired contrast (though often, it does).

Histogram equalization produces acceptable results for most common images, but it is sometimes sensitive to small background color noise, causing it to be amplified in the result. This happens when the image has a very sparse range of values in its histogram so HE tries to shift these sparse values across the entire range.

2.2 Contrast Stretching

In contrast stretching, the image histogram is tri-partitioned; any pixels in the lowest partition are assigned value zero, any pixels in the highest partition are assigned value 255, and the pixels in the middle are proportionally stretched to fit the entire histogram. (This is like creating a transfer function with a corresponding flat band, then a narrower slope, then another flat band.) The challenge of contrast stretching is in determining how to partition the histogram.

The usual version of contrast stretching is also limited to only one middle rising band defining the contrast gain. As we will later see, OCTM generalizes this to have multiple possible sections of increased and decreased contrast gain.

2.3 Hardware Solution

To combat low-light photography is difficult but there are many solutions to this such as prolonging exposure time, using flash or higher ISO. These can be pretty limited in what they achieve without significant cost (waiting a very long time for exposure), when as we will later see there are some fairly good software methods that sacrifice a little bit of quality for an enormous benefit in cost and time.

3 Optimal Contrast Tone Mapping

OCTM (Optimal Contrast Tone Mapping) is a method which argues that qualitatively, the cumulative density function should be modified to have higher slopes where the tones of the images are concentrated, and flatter slopes in the range where the image does not contain many tones. In essence, this provides sharper distinctions of colors that appear more often, and squishes together all the colors that do not appear as often with the hope that due to their scarcity, the viewer will not notice.

Contrast, in the OCTM sense, is about providing as many of these sharp slope gains in image histograms as possible to the best locations. The definition can now be formalized mathematically.

3.1 Contrast

To maximize contrast, one thing we could try would be to maximize color differences between adjacent objects, but this alone is not enough because there are many objects competing for only finitely many available shades of color. We can look at this as a resource allocation problem where we prioritize which colors to enhance contrast for; the idea of OCTM would be to take the colors that occur more frequently.

From this idea it is clear that we need to look at the histogram of the image. Let p_i denote the occurrence of tone i in the image; it does not matter whether p_i is considered to be the number of occurrences or the probability of occurrence, as long as its ratios to p_j, $j \neq i$ are consistent.

Our contrast definition $G(\boldsymbol{s})$, and henceforth objective function to be maximized, is defined as

$$G(\boldsymbol{s}) = \sum_{i=0}^{L-1} p_i s_i \tag{1}$$

where $s_i \in \{0, 1, \cdots, 255\}$ is the grey level gain between adjacent tones s_i and s_{i+1} in the input histogram. For example, if input level 0 maps to output level 0 and input level 1 maps to output level 2, $s_0 = 2$. Note that if $s_i = 1$ for all i then we have histogram equalization, so OCTM is a generalization of HE.

The final version of OCTM should have some constraints added for practical purposes. Clearly, the maximum of $G(\boldsymbol{s})$ in its current form is unbounded so constraints must be added to account for the fact that only finitely many shades of color exist. Another one of the challenges of building a good contrast enhancement algorithm is striking a balance between strong contrast and other visually desirable traits such as smoothness, so having proper constraints to mediate this is essential. Ideally, each constraint should be linear, since there exist many efficient polynomial-time algorithms to solve linear programming.

3.2 General Formation of OCTM

The most general form of OCTM is given below. The two constraints will be explained in subsequent sections.

$$
\begin{aligned}
\max_{\boldsymbol{s}} \quad & G(\boldsymbol{s}) = \sum_{j=0}^{L} p_j s_j && \text{(2a)}\\
\text{subject to} \quad & \sum_{j=0}^{L} s_j \leq L - 1, && \text{(2b)}\\
& \frac{1}{d} \leq s_j \leq u, 0 \leq j < L. && \text{(2c)}
\end{aligned}
$$

3.3 Transfer Function Constraints

First, considering the definition of $G(\boldsymbol{s})$, the definition the transfer function $T : \{0, 1, \cdots, 255\} \rightarrow \{0, 1, \cdots, 255\}$ from input to output is given as

$$
T(i) = \sum_{i=0}^{L-1} s_i \tag{3}
$$

Practically, the transfer function should not exceed the output dynamic range. Thus, we must have that $\sum_{i=0}^{L-1} s_i \leq L - 1 = 255$. Moreover, it is desirable to have that tones are not reversed (i.e. if $i > j$, then $T(i) \geq T(j)$); the nature of the maximization problem guarantees $s_i \geq 0 \forall i$, so following from this the definition of T implies its monotonicity (in the same manner as HE's transfer function implies this as well). This constraint alone guarantees a finite dynamic range, but is not enough and will give rise to the following trait in the output images:

Proposition 1 (Wu, 2011): Suppose that the most frequently occurring tone in an image occurs at index k, i.e. $p_k = \max p_i : 0 \leq i < L$. The maximum contrast gain $G(\boldsymbol{s})$ is achieved by $s_k = L - 1$ and $s_j = 0$, $j \neq k$.

The proof is given in the lecture slides/paper. The statement essentially says that the transfer function is maximized by creating a step transfer function between the tone that most frequently occurs and the one before it; given that there are no other constraints on $G(\mathbf{S})$ at the moment, it makes perfect sense that a binary thresholder would maximize contrast. This is perceptually unpleasing so additional constraints are be added to limit it.

3.4 Tone Distortion

The problem of preventing very obvious thresholding in images can be partially addressed by enforcing a minimal tone increase in a given histogram interval. For example, we may decide that every five input levels, there must be one output level increase (i.e. no more than five input grey levels are mapped to the same output grey level). This minimal tone increase ensures that within a given histogram range, there is *at least some nonzero level of dynamic range* so that details are not fully lost, to give the image a more natural appearance.

Minimal tone increase be mathematically formulated as:

$$
\sum_{j \leq i < j+d} s_i \geq 1, 0 \leq j < L - d \tag{4}
$$

where d is the minimum increase interval. Consider first the case where $s_i \in \mathbb{Z}$ (which is the most natural representation, since image levels are represented as integers). Then this becomes an integer programming problem which is NP-Hard if we should try to solve it naively. However, we believe that this problem may be modeled as a directed acyclic graph which could provide enormous time complexity speedup using dynamic programming, but we have not investigated this in depth so we cannot verify it.

Now consider allowing $s_i \in \mathbb{R}$. This sort of constraint can actually be converted to a linear program by adding an individual constraint for each interval $\{0, 1, ..., d\}, \{1, 2, ..., d+1\}, \{L-d-1, L-d, ..., L-1\}$ (the constraint matrix would have the form of a Toeplitz matrix with ones and zeros). Alternatively, the constraint can be relaxed as

$$s_j \geq \frac{1}{d}, 0 \leq j < L \tag{5}$$

where instead of enforcing a minimum increase d over an interval, we have averaged that d across each element in the interval and requested that each element increase by at minimum that average amount. This makes the problem formulation simpler and less computational intensive.

Also note that now, since $s_i \in \mathbb{R}$, the transfer function needs to be discretized so we round the result as follows:

$$T(i) = \left\lfloor \sum_{i=0}^{L-1} s_i + 0.5 \right\rfloor \tag{6}$$

By enforcing a minimum tone increase in the output image, we have now fixed half the tone distortion problem.

	OCTM from Proposition 1 (L-1=8)					mode			
p	0.1	0.1	0.2	0.05	0.2	0.25	0.05	0.025	0.025
s	0	0	0	0	0	8	0	0	0
	OCTM from Proposition 2 (L-1=8, d=4)								
p	0.1	0.1	0.2	0.05	0.2	0.25	0.05	0.025	0.025
s	0.25	0.25	0.25	0.25	0.25	6	0.25	0.25	0.25

Figure 1: Example of optimal allocations, (top) without 5, (bottom) with 5. Allocate the minimum 1/d to each s then fill the s with the highest corresponding p with the remaining amount.

Proposition 2: Suppose image I has a unique modal (most frequently occurring) tone at index k, similar to the premise of Proposition 1. The maximum contrast gain $G(\boldsymbol{s})$ is achieved by $s_k = (L-1)(1-\frac{1}{d})$ and $s_j = \frac{1}{d}$, $j \neq k$, $L, d > 1$.

Proof: First, note that $\sum_{\text{all}s} s_j = L-1$, so there is no way to add more contrast individually. Now we show this is the optimal allocation by contradiction. Suppose that in $\exists j$ such that $s'_j = \frac{1}{d} + \epsilon$ for which $G(\boldsymbol{s'}) > G(\boldsymbol{s})$ is maximal. Then we have $p_k s_k + p_j s_j = p_k(L-1)(1-\frac{1}{d}) + p_j \frac{1}{d} > p_k[(L-1)(1-\frac{1}{d}) - \epsilon] + p_j(\frac{1}{d} + \epsilon) = p_k s'_k + p_j s'_j$, since $0 > -p_k + p_j$. This contradicts the original assumption, since we can redistribute s'_j from p_j to p_k to increase contrast (note that all distribution of the other tones $p_i, i \neq j, i \neq k$ are untouched).

Corollary 1: This linear programming problem can be solved by finding the max of the vector $\mathbf{p}$ and allocating values of $\mathbf{s}$ to the indices where the maximum values of $\mathbf{p}$ occur, running in $O(L)$.

Proposition 2 essentially states that after allocating the minimum quota of $\boldsymbol{s}$ to every non-modal tone, the remaining quota should be devoted to the modal tone; this is only a marginal improvement from maximizing $G(\boldsymbol{s})$ without constraint 2c. Experimentally, we have observed that this alone is not a very good method because often the resulting image looks like a failed attempt at thresholding, especially when d is large.

From these observations, it seems like introducing an upper bound on the vector $\mathbf{s}$ could help by suppressing the thresholding effect, and allowing multiple different tones to have a fair share of increased contrast. The following reformulation with an upper bound u is proposed:

$$\max_{\boldsymbol{s}} \quad G(\boldsymbol{s}) = \sum_{j=0}^{L} p_j s_j \tag{7a}$$

$$\text{subject to} \quad \sum_{j=0}^{L} s_j \leq L-1, \tag{7b}$$

$$\frac{1}{d} \leq s_j \leq u, 0 \leq j < L. \tag{7c}$$

Proposition 3: Suppose image I has most frequently occurring tones of probability $p_a \geq p_b \geq p_c \geq ... \geq p_k \geq ...$, and the tone jump s_k is upper bounded by u. There is a long and messy expression for the maximum contrast gain, which we omit but provide an example of later.

Corollary 2: This linear programming problem is slightly more complicated than proposition 2, but can still be solved simply by sorting the vector $\mathbf{p}$ and allocating values of s to the indices where the maximum values of $\mathbf{p}$ occur, running in $O(L \log L)$.

	OCTM from Proposition 1 (L-1=8)					mode			
p	0.1	0.1	0.2	0.05	0.2	0.25	0.05	0.025	0.025
s	0	0	0	0	0	8	0	0	0
	OCTM from Proposition 2 (L-1=8, d=4)								
p	0.1	0.1	0.2	0.05	0.2	0.25	0.05	0.025	0.025
s	0.25	0.25	0.25	0.25	0.25	6	0.25	0.25	0.25
	OCTM from Proposition 3 (L-1=8, d=4, u=2)								
p	0.1	0.1	0.2	0.05	0.2	0.25	0.05	0.025	0.025
s	0.25	0.75	2	0.25	2	2	0.25	0.25	0.25

Figure 2: Example of optimal allocations, (top) without 2c, (middle) with 2c, (bottom) with 7c. With 7c, start by allocating 0.25 to each, then upgrade the bins of largest p values to 2 until constraint 2c is reached.

The upper bound fixes the threshold-like appearance issue, so this completes the most general formulation of OCTM. Visual quality is already sufficiently good using only this. There are some more optional constraints for special cases which will be discussed in the next section; from experimental observation, they are not very useful in most circumstances. They will be discussed further in each corresponding section.

The constraints from here on are not part of the main set of constraints for OCTM in this project. In particular, note that the proofs from above do not hold in these next conditions. They may be added for more specific applications.

3.5 Gamma Correction

One extra feature that can be embedded into OCTM is Gamma Correction, in the form of the following constraint:

$$\sum_{i=0}^{L-1} \left| (L-1)^{-1} \sum_{j=0}^{i} s_j - [i(L-1)^{-1}]^{\gamma} \right| \leq \Delta \tag{8}$$

where γ is the gamma correction factor and Δ is the allowed deviation from the actual gamma curve. Note that this constraint does not appear to be linear in $\boldsymbol{s}$ because of the absolute value within summation, but it can be made linear by considering all 2^L possible outcomes of the elements in the summation (if the first term is greater, or the second term is greater) and generating an individual constraint for each one. With $L = 256$ this is clearly intractable, so we consider the following alternative using the triangle inequality:

$$\left|\sum_{i=0}^{L-1}(L-1)^{-1}\sum_{j=0}^{i}s_j - [i(L-1)^{-1}]^{\gamma}\right| \leq \sum_{i=0}^{L-1}\left|(L-1)^{-1}\sum_{j=0}^{i}s_j - [i(L-1)^{-1}]^{\gamma}\right| \leq \Delta \tag{9}$$

This is upper bounded by (8), so it is guaranteed to also be less than Δ. To implement this in MATLAB, it must then be expressed as two individual constraints:

$$\sum_{i=0}^{L-1}(L-1)^{-1}\sum_{j=0}^{i}s_j \leq \Delta + \sum_{i=0}^{L-1}[i(L-1)^{-1}]^{\gamma} \tag{10}$$

$$-\sum_{i=0}^{L-1}(L-1)^{-1}\sum_{j=0}^{i}s_j \leq \Delta - \sum_{i=0}^{L-1}[i(L-1)^{-1}]^{\gamma} \tag{11}$$

which can be seen to be linear in $\boldsymbol{s}$ by reversing the order of summation as follows:

$$\sum_{i=0}^{L-1}s_i\sum_{j=0}^{L-i-1}(L-1)^{-1} \leq \Delta + \sum_{i=0}^{L-1}[i(L-1)^{-1}]^{\gamma} \tag{12}$$

$$-\sum_{i=0}^{L-1}s_i\sum_{j=0}^{L-i-1}(L-1)^{-1} \leq \Delta - \sum_{i=0}^{L-1}[i(L-1)^{-1}]^{\gamma} \tag{13}$$

The inner summation on the left-hand side and the gamma summation on the right-hand side can be precomputed efficiently.

3.6 Mean Shifting

To prevent the image mean from being modified beyond a certain limit Δ_μ, the following constraint can be implemented:

$$\left|\sum_{i=0}^{L-1}p_i\sum_{j=0}^{i}s_j - \sum_{i=0}^{L-1}p_i i\right| \leq \Delta_\mu \tag{14}$$

which can be seperated into two constraints without absolute value in the same way as the Gamma Correction constraints were done. The first term computes the mean of the candidate output OCTM image, and the second term is the mean of the input image.

In practice, a high mean shift may be desirable for images under extremely low lighting so this constraint is not helpful in these purposes. Perhaps it would be useful if OCTM tries to overexpose an image (making it too bright) by fine tuning Δ_μ to something small.

3.7 Color Images

The formulation of OCTM from before was focused on application to greyscale images. To enhance color images, one thing that should be emphasized is that OCTM **should not** be applied to each of the channels R,G,B separately; this will produce false colors in the resulting image. Instead, the image representation should be converted to a colorspace with a lightness channel such as HSV, then apply OCTM to the lightness (V) channel.

The HSV representation was designed to use V to distinguish a given H,S between its most possible vividness and darkness (see the figure). This is ideal for the OCTM application, since we can argue that high contrast will imply more vivid colors as high-lighting conditions exposes more vividness in a scene.

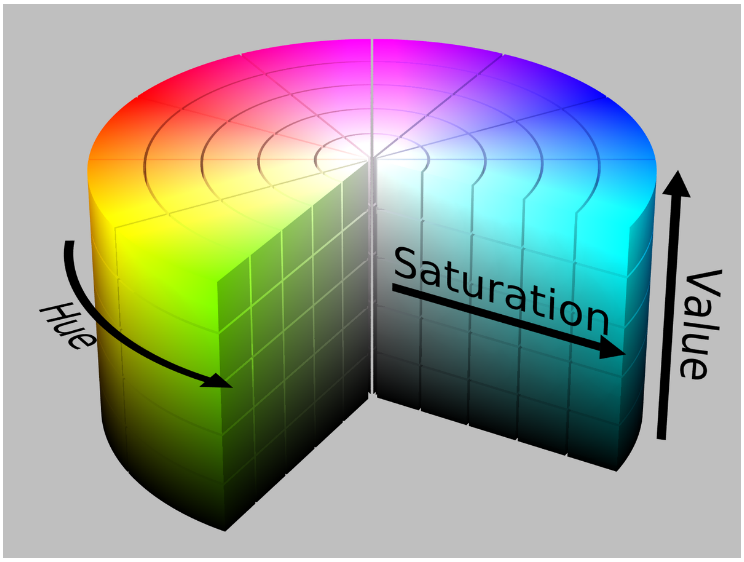


Figure 3: Visual representation of the HSV colorspace.
(Wikipedia)

4 Experimental Results

Figure 4 and 5 present some sample images that are enhanced by the OCTM technique in comparison with those produced by histogram equalization, and contrast stretching. In some images, HE over-stretches the contrast and make the image visually unpleasant. Contrast stretching method (built-in MATLAB function) works well in some of the images.

Subplot **(c)** and **(d)** in Figure 4 and 5 also present some sample images which the low contrast images are firstly reorganized in Bayer pattern and then demosaiced using the algorithm discussed in **Project 1** (PCSD) and finally enhanced by OCTM.

By comparing the original image and demosaiced image without enhancement by OCTM, although the PCSD demosaicing algorithm is much better than bilinear interpolation, however, artifacts and discolorations can still be seen in the demosaiced images, especially in the high frequency regions. Notice also in the OCTM enhanced images, the artifacts are even more pronounced. Since the contrast of the image is enhanced, some artifacts and discolorations are not particularly visible in the low contrast images, became much more apparent and visually unpleasant, as shown clearly in Figure 6.

For practical applications, it is necessary to fine tune the parameters d,u (and Δ, Δ_μ, γ should they be used) to get the best possible visual results. The guideline for selecting u is based on how dark the image is. Very dark images would benefit greatly from a higher u value, which allows more contrast gain per grey level. On the other hand, d is selected with mostly a guess-and-check approach, noting that images with lots of subtle details in many different regions will suffer lots of information loss if d is too large. On the other hand, we did not notice too much benefit from adjusting the Δ, Δ_μ, γ parameters.

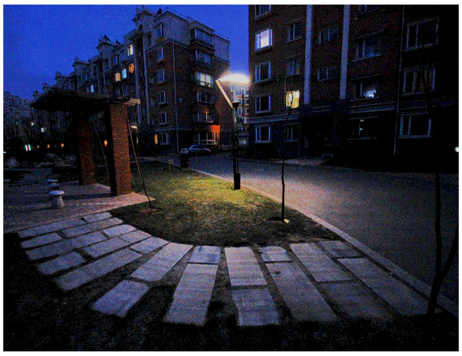
Figure 7 and 8 present some OCTM enhancement result with different parameters. The parameters where tuned based on the contrast metric, our perception, output transfer function from OCTM, as well as the histogram of the V of the HSV plane. For example, from Figure 9, it is clear that the transfer function from **(b)** will do a better job of stretching the low contrast **im2.png**; Similarly, in Figure 10, the histogram of V plane from **(b)** looks promising due to the fact that the histogram is stretched to occupy the whole 0-255 intensity level span while still have a reasonable amount of non-empty intensity level bins; By using the Contrast metric tool defined in the paper, the results coincide the our observations, **(b)** has the highest contrast (shown in the title of each subplot).

Following similar principles, Figure 11 and 12 show that (c) and (d) are good picks for the d parameter. As discussed before, although changing the d parameter will not always result in a drastic difference in terms of contrast, it is still worth the tuning process.

(a)

(b)

(c)



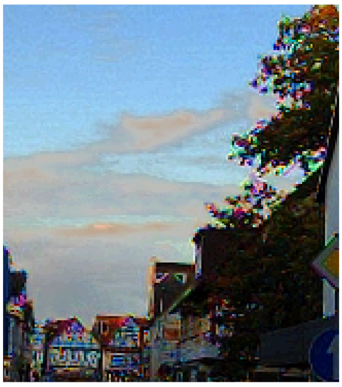
(d)

Figure 4: Comparison of different methods on im1. (a) Contrast Stretching. (b) HE. (c) OCTM. (d) PCSD then OCTM.

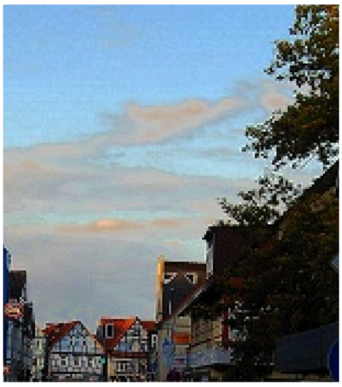
(a) (b)

(c) (d)

Figure 5: Comparison of different methods on im2. (a) Contrast Stretching. (b) HE. (c) OCTM. (d) PCSD then OCTM. Note that PCSD has minor artifacts around the tree.



(a)



(b)

Figure 6: Closer look at the artifacts from PCSD and pronounced by OCTM contrast enhancement.

(a)

(b)

(c)

(d)

Figure 7: Comparison of different upper-limit parameter on im2 (d=50). (a) u=2. (b) u=7. (c) u=15. (d) u=30.

(a)

(b)

(c)

(d)

Figure 8: Comparison of different d parameter on im2 (ul=7). (a) d=2. (b) d=5. (c) d=20. (d) d=50.

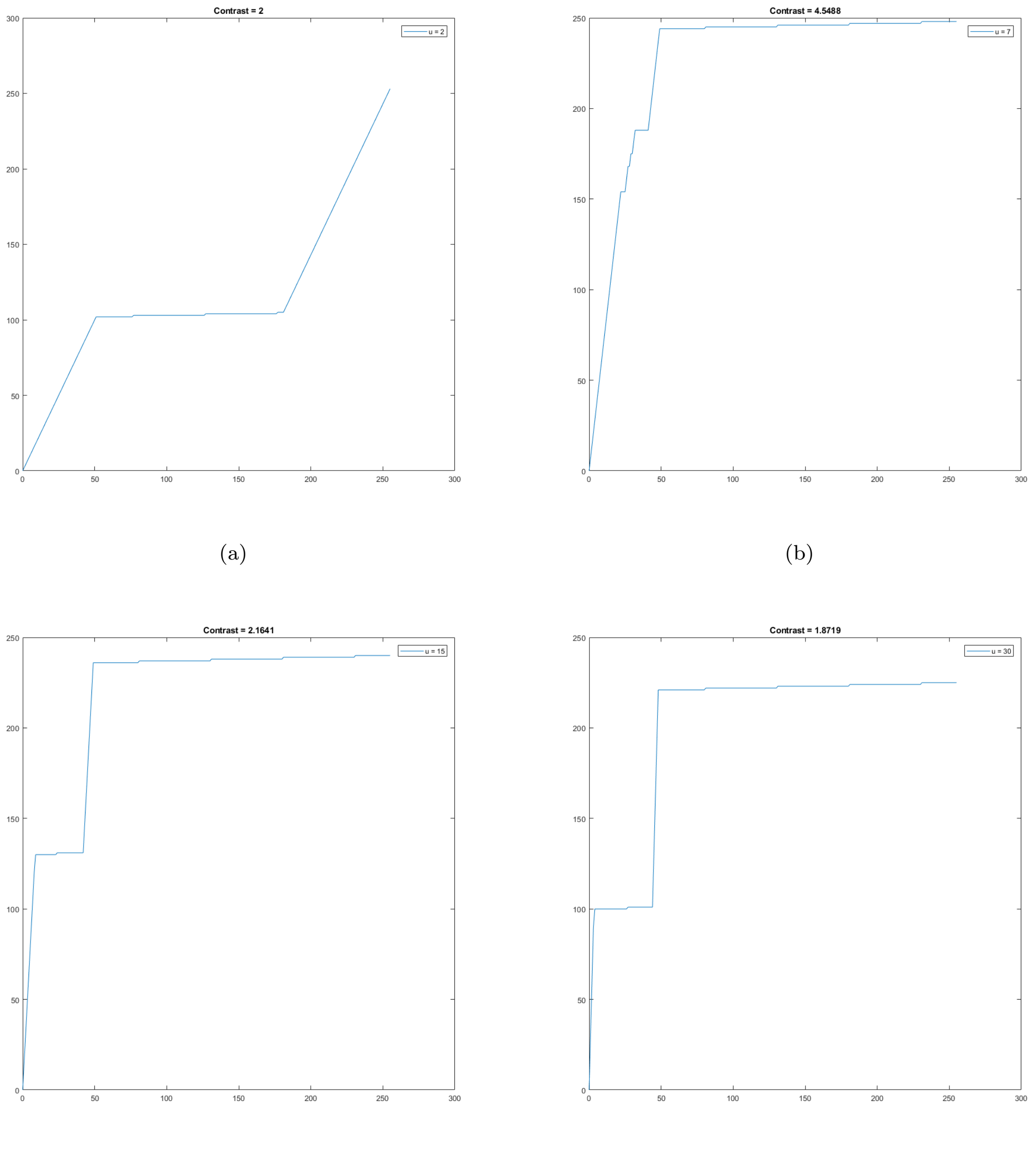
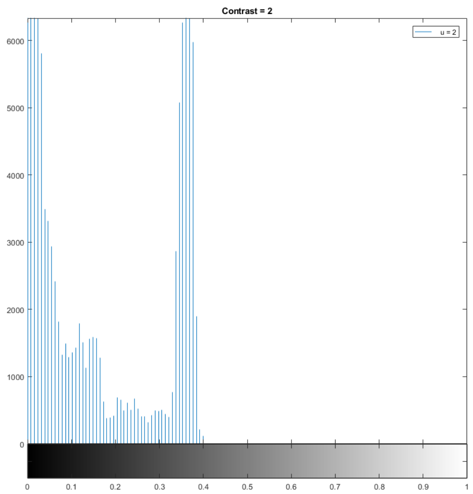
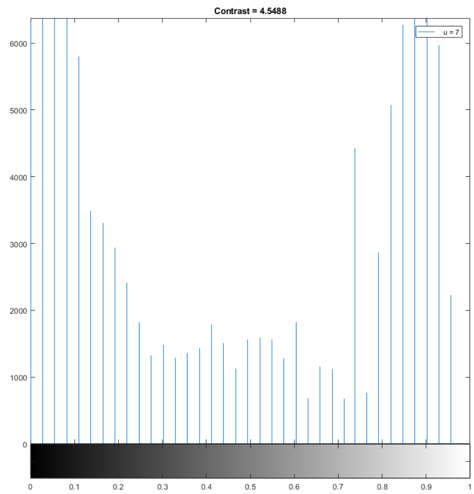


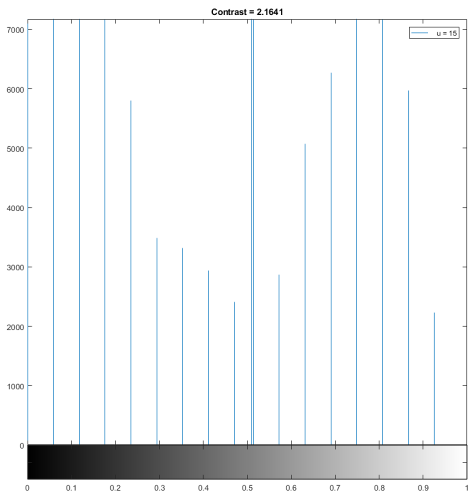
Figure 9: OCTM output transfer function of different upper-limit parameter on im2 (d=50). (a) u=2. (b) u=7. (c) u=15. (d) u=30.



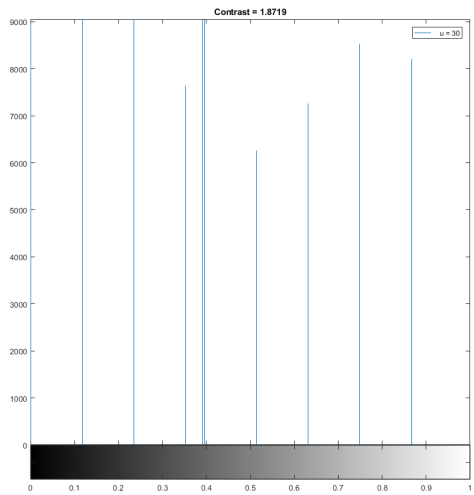
(a)



(b)

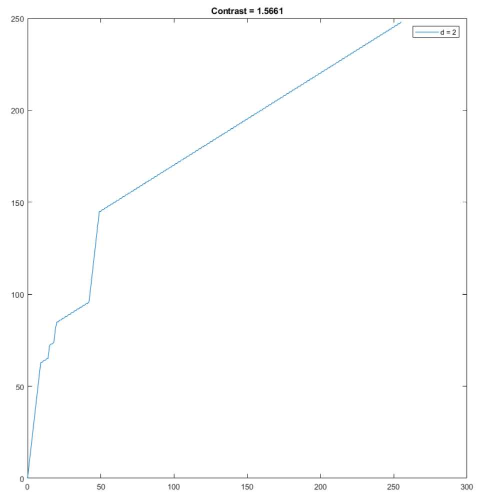


(c)

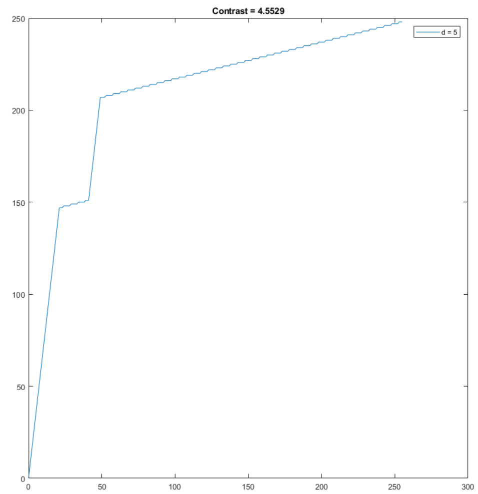


(d)

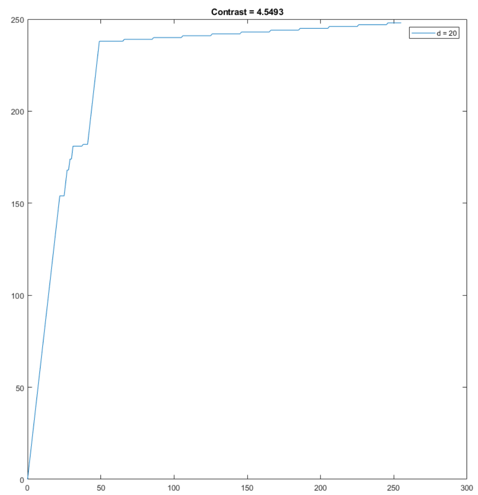
Figure 10: OCTM output histogram of V plane of different upper-limit parameter on im2 (d=50). (a) u=2. (b) u=7. (c) u=15. (d) u=30. One can see that larger the u parameter, bigger the gap bewteen non-empty intensity level bins.



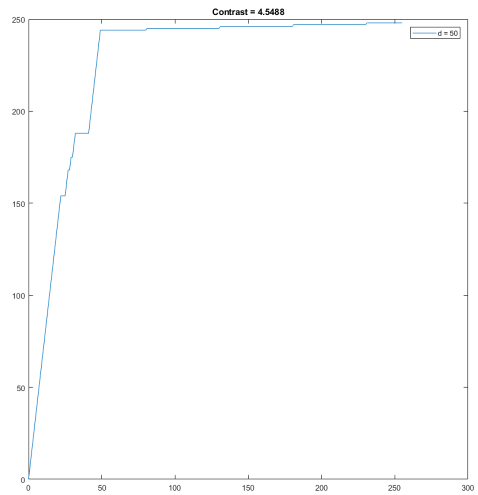
(a)



(b)

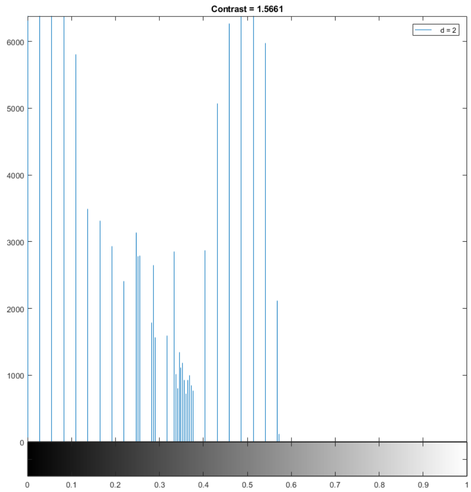


(c)

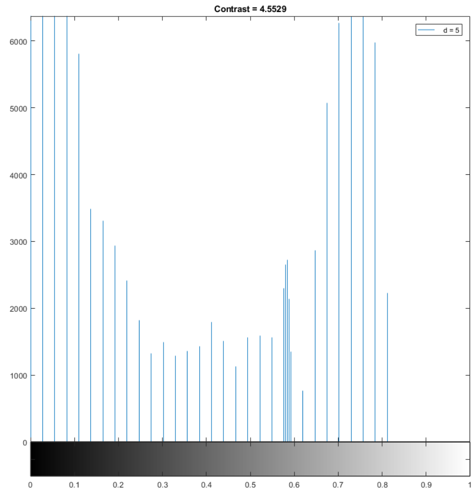


(d)

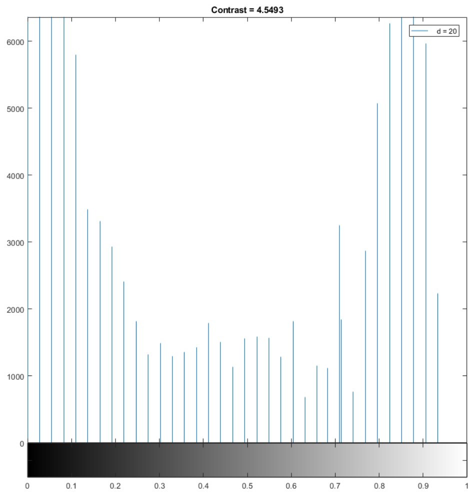
Figure 11: OCTM output transfer function of different d parameter on im2 (u=7). (a) d=2. (b) d=5. (c) d=20. (d) d=50.



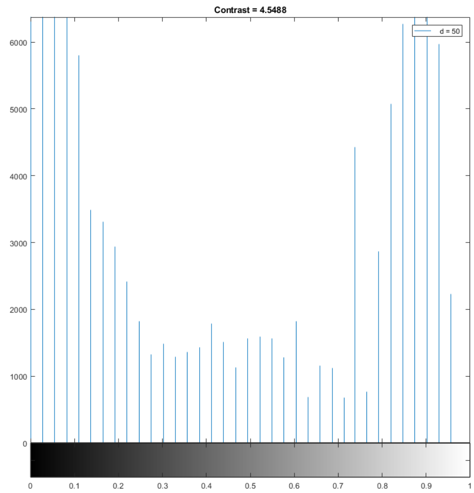
(a)



(b)



(c)



(d)

Figure 12: OCTM output histogram of V plane of different d parameter on im2 (u=7). (a) d=2. (b) d=5. (c) d=20. (d) d=50. Note that increasing d forces larger minimal increase, thus decreasing more tones to increase by u. Increasing d also makes the $\frac{1}{d}$-allocated tones increase by a much smaller amount, thus when rounded with high probability they will simply map to the value of the last u-allocated tone.

5 Summary

In summary, we see that OCTM can be employed to generate more realistic looking results in some of the cases we presented. It is important to note that there is not a "one-fit-all" algorithm that performs best for each image; each algorithm has its own benefits for certain types of images. Moreover, what each viewer perceives as the best algorithm will be subjective so there is no perfect measurement of how good "contrast" is; this can lead to disagreements on optimal tuning values (such as d and u in OCTM).

Additionally, the problem becomes more challenging when the original image generated from the mosaic pattern. The output will never be truly as good as the result without mosaic since there was less information to begin with.

Future algorithms may employ more sophisticated techniques such as considering context or image locality. For example, considering the value of a pixel's neighbors provides much more detail about how much contrast is necessary to make it stand out more.