The Effects of Ellipsoidal Volume on Differential Metamer Generation

Joseph A. Boyle

May 5, 2017

Abstract

We attempt to maximize and minimize the volumes of separating ellipsoids to study the impact of ellipsoidal volume on differential metamer generation. Despite having a similar accuracy as randomly sampled points of unconstrained volume, ellipsoids with minimized volumes perform better visually.

1 Introduction

Differential Metamers [1] are 6-dimension tuples {$\bar{c}$, $\hat{i}$} in color space (where $\bar{c}$ is [R, G, B], and $\hat{i}$ is a 3-dimensional unit vector), such that for some α, the alternating of $\bar{c}$ and $\bar{c} + \alpha\hat{i}$ is indistinguishable to the human eye, but distinguishable by a camera. Given a mapping $D(\bar{c}) \Rightarrow \alpha\hat{i}$, for all $\bar{c}$ in the 8-bit RGB color space, we can encode any image with a binary message via the following mapping M, where $\bar{c}$ is the pixel at some x, y in the image, and $message_i$ is the bit at a region of the image in which the pixel is:

$$M(\bar{c}) = \begin{cases} \bar{c}, if message_i = 0 \\ \bar{c} + D(\bar{c}), if message_i = 1 \end{cases}$$

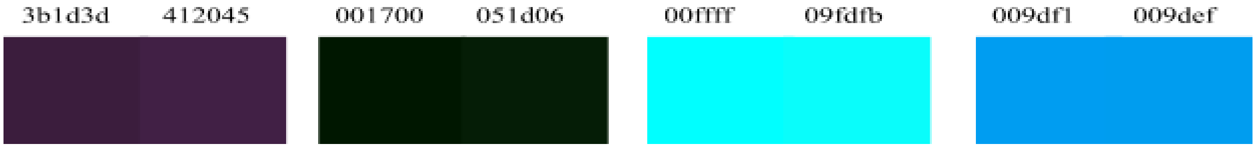

Figure 1: Four colors and their corresponding pair. The right two are "visually good," while the left two are not.

Given a base image and an image with an embedded image based on the mapping M, we are able to transmit messages in plain sight without specialized hardware or the transmission being obvious. Visual MIMO [2] is a system which implements this encoding strategy, but requires an abundance of differential metamers, such that the system can decode messages quickly and without error.

1.1 Previous Work and Methods

Labelling a set of pairs is a two step process. First, we record a color $\bar{c}$ in one frame, and a checkerboard pattern of $\bar{c}+\hat{i}$ and $\bar{c}$ in the next (ie: $message = 10101010...$), alternating at roughly 4 FPS, with a camera, which samples at twice the framerate of the display [3]. By sampling at twice the display's framerate, we eliminate the problem of a rolling shutter producing "dirty"

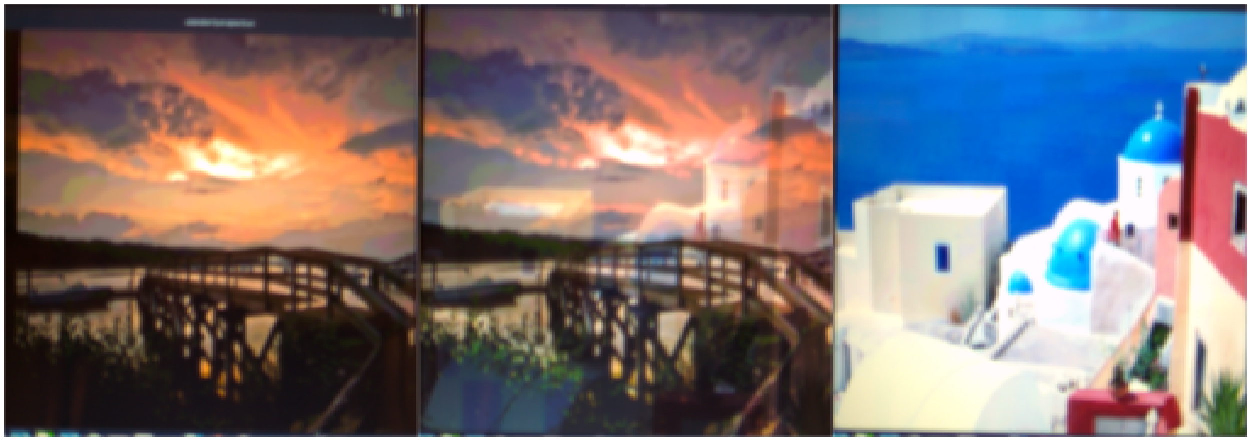

Figure 2: Three subsequent frames, where the middle (dirty) frame contains bleeding from the surrounding (clean) frames.

images (Figure 2), as we can manually remove all frames which occur on a shutter close. Next, we can subtract the obtained "clean" frames, and extract the message [1, 4]. We label all pairs which have a Bit Error Rate (BER) equal to zero as *good*$_{camera}$, and all pairs with a higher BER as *bad*$_{camera}$.

$$\text{BER} = \frac{\text{bits incorrectly recovered}}{\text{total bits in } message}$$

Next, we view each of the previously generated videos manually. If there appears to be a flicker or other artifacts during the alternation of the frames, the pair is labelled as *bad*$_{visual}$, or *good*$_{visual}$ otherwise. The set of *good* points, then, is $good_{visual} \cap good_{camera}$, and *bad* is $all - good$.

2 Differential Metamer Generation

Previous work [1, 5] has led to the development of a framework by which we can find new color pairs which potentially are differential metamers via the generation of separating ellipsoids, provided we have pre-labelled sets of color pairs, *good* and *bad*, which are indiscernible to human vision tests but not to cameras, and those that fail either test, respectively.

The sets of colors are broken into k smaller clusters [5]. For each of the k clusters, we then attempt to find an ellipsoid ε such that it includes all of the N *good* points and none of the M *bad* points. This is a convex optimization problem [6] in which we attempt to find a $P \in S^6$, $\bar{q} \in R^6$, and $r \in R$, such that $P \succeq 0$, and satisfies the following conditions, where x are positive (*good*) points, and y are negative (*bad*) points:

$$\begin{aligned} \bar{x}_i^T P \bar{x}_i + \bar{q}^T \bar{x}_i + r &>= 1 - u(i), \text{ for } i = \text{i, ..., N} \\ \bar{y}_i^T P \bar{y}_i + \bar{q}^T \bar{y}_i + r &<= v(i) - 1, \text{ for } i = \text{i, ..., M} \end{aligned} \tag{1}$$

We attempt to minimize $\sum_i^N u(i) + \sum_i^M v(i)$ in solving the problem. In doing so, we allow a small margin of error in our ellipsoid calculation, which is better than failing if no ellipsoid can totally separate the points. With a P, $\bar{q}$, and r, we can sample points randomly and determine if they are within any of the k ellipsoids.

2.1 Ellipsoidal Volume

We empirically choose a k such that each ellipsoid, ε_k, spans some ideal number of points within the training set. The generated ellipsoid seeks to simply enclose the good points, without guaranteeing upper or lower boundaries on their volume. In effect, any two ellipsoids, ε_1 and ε_2, could separate an entire cluster and yet have largely differing volumes. Typically, these volumes are fairly consistent, but there is no attempt to minimize nor maximize them.

Since P is a symmetric, positive definite matrix, the volume of an ellipsoid ε is proportional to the determinant of P [6], where γ is the volume of the unit ball of R^6, ie: $\frac{\pi^3}{6}$:

$$\begin{aligned}\text{Volume} &\propto \gamma * |P|^{\frac{1}{2}} \\ &\Rightarrow log(\gamma) + \frac{1}{2}log(|P|) \\ &\propto log(|P|)\end{aligned} \tag{2}$$

We take the logarithmic volume to obtain a concave function with respect to P; that is, the volume of the ellipsoid increases with respect to the $|P|$. We can maximize the volume by maximizing the log($|P|$). To minimize the volume, which is inversely proportional to $|P^{-1}|$, we can minimize -log($|P|$):

$$\begin{aligned}\frac{1}{\text{Volume}} &\propto |P^{-1}| \\ &= \frac{1}{|P|} \\ &\Rightarrow -log(|P|)\end{aligned} \tag{3}$$

Ideally, we would set upper-bounds on the volume of an ellipsoid, and guarantee that any ellipsoid ε had a volume strictly less than that upper bound. Unfortunately, constraints of the format $f \leq g$,where f is a concave function, are violations of the Disciplined Convex Programming (DCP) ruleset (which is utilized by the system which solves the problem). There is no way to formulate our problem such that it is concave with respect to P. Instead, we can experiment with the effects of ellipsoidal volume by maximizing and minimizing the volume, to observe extremas. Thus, when minimizing the volume, we use the system:

$$\begin{aligned}\text{minimize(-log(}|P|) + (\sum_{i}^{N} u(i) + \sum_{i}^{M} v(i))) \\ \text{Subject to:} \\ \bar{x}_i^T P \bar{x}_i + \bar{q}^T \bar{x}_i + r >= 1 - u(i),\ \text{for } i = \text{i, ..., N} \\ \bar{y}_i^T P \bar{y}_i + \bar{q}^T \bar{y}_i + r <= v(i) - 1,\ \text{for } i = \text{i, ..., M}\end{aligned} \tag{4}$$

Maximizing the volume uses the system:

$$\begin{aligned}\text{maximize(log(}|P|) - (\sum_{i}^{N} u(i) + \sum_{i}^{M} v(i))) \\ \text{Subject to:} \\ \bar{x}_i^T P \bar{x}_i + \bar{q}^T \bar{x}_i + r >= 1 - u(i),\ \text{for } i = \text{i, ..., N} \\ \bar{y}_i^T P \bar{y}_i + \bar{q}^T \bar{y}_i + r <= v(i) - 1,\ \text{for } i = \text{i, ..., M}\end{aligned} \tag{5}$$

2.2 Current Challenges

Despite experimentation with clustering methods [5], the generation time of an ellipsoid is dominated by the number of points closed within and their relative disorder. That is, as the number of intermingled points increases, the running time to compute P, $\bar{q}$, and r increases non-linearly. This also has a spatial cost, such that for training data of approximately 15,000 points split among 20 ellipsoids, 8GB of RAM becomes insufficient.

Besides being slow to generate, points within the ellipsoid can be quite slow to label, to the tune of roughly 30 minutes per 1,000 points. Ideally, every color pair generated by an ellipsoid would be a differential metamer so as to alleviate the need to manually label points. Practically, this isn't possible, and so we either need to manually verify large amounts of data or be able to use unlabelled points within a certain degree of accuracy.

2.3 Speed Improvements

We employ a slight optimization over the ellipsoid generation technique implementation [6] in 2.1.1, so as to remove $N + M - 2$ inequality constraints, where $good$ is an Nx6 matrix, and bad is an Mx6 matrix:

$$\begin{aligned} \Big(\sum(good^T * P) \cdot good^T\Big) + good^T * q + r &>= 1 - u \\ \Big(\sum(bad^T * P) \cdot bad^T\Big) + bad^T * q + r &<= v - 1 \end{aligned} \tag{6}$$

With the previous implementation, generating 5 ellipsoids from $8,400$ testing points took 115 minutes, growing nonlinearly as the number of points grew. Under similar conditions, generating 5 ellipsoids from the same starting data took under two minutes via this implementation.

3 Proposed Works

We seek to observe the effects of maximizing and minimizing the ellipsoid volume. Specifically, we will design two experiment groups: ellipsoids generated with their volume being bound toward an extrema ($\varepsilon_{experimental}$), and ellipsoids generated without any modification to their volume ($\varepsilon_{control}$). We will take a subset of the points generated within $\varepsilon_{control}$ and label them as described in 1.1. These labelled pairs will be called $D_{labelled}$. These same points, with their labels removed, will constitute $D_{unlabelled}$.

We will then examine the BER of message recovery for images embedded using each of our data sets. We expect the BER (lower is better) of images embedded using the color pairs in $D_{unlabelled}$ to be quite high with quite noticeable screen flicker, while the BER of images embedded using the color pairs in $D_{labelled}$ should be low, with very little image flicker. The images embedded using points from the $\varepsilon_{experimental}$ are expected to have BER and screen flickers falling between these two ranges. To test the versatility of color pairs, we embed a checkboard message in a variety of images that have regions of varying texture and color (Figure 3). Videos consist of one frame each of the base and embedded images [1, 4], displayed at 4 FPS, while the camera samples at 8 FPS. All tests use an α of 10.

For consistency, all experiments were conducted using a Basler acA2040-90uc-CVM4000 camera, which had its white-balance adjusted to the same image for every experiment (image #7 along the top-row in Figure 3) and was position approximately 1.5 meters from the screen.

Figure 3: The fourteen images used to test BER rates.

3.1 Control Data Sets

Beginning with a set of 166 differential metamers ($D_{initial}$), five ellipsoids were generated, to be used to generate the training data for our experiments. From these five ellipsoids, 8,400 points ($D_{training}$) were evaluated, and labelled as good and bad. This set was used to generate all future ellipsoids for both the control and experimental sets.

The ellipsoids come from three separate generations, each of which had a small subset of its total points evaluated, and were generated from all previous training points. The three generations had $k = 5$, $k = 20$, and $k = 20$ clusters, respectively, as the size of the training data quickly grew. After sampling 18,600 points total ($D_{training}$ included), we find 2,566 good pairs ($D_{labelled}$). The unlabelled dataset ($D_{unabelled}$) is simply the union of the good and bad pairs that we classified, but with their labels removed.

3.2 Experimental Data Sets

We attempt to, at $k = 20$ and $k = 80$, produce ellipsoids with their volumes minimized and maximized. The k-values were chosen such that we can see how a larger number of ellipsoids spanning the same set of points affects the extrema of their volumes. As a baseline, for each k-value, we also compute k ellipsoids without minimizing or maximizing the volume, such that we can observe the standard deviation. There are four resulting sets of 18,600 color pairs each, then, used to embed a message into images:

1. Maximum volume at $k = 20$
2. Minimum volume at $k = 20$
3. Maximum volume at $k = 80$
4. Minimum volume at $k = 80$

4 Results

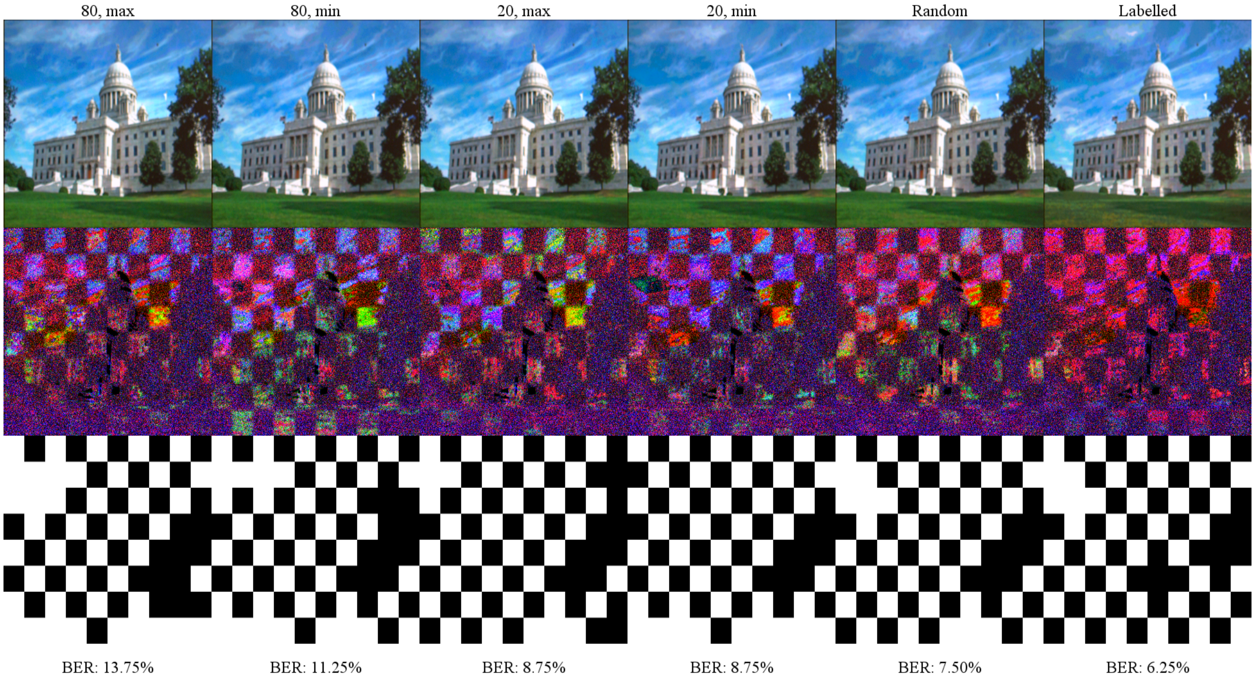

Figure 4: Side by side comparisons of image 4 (Figure 3) being embedded and decoded with the various data sets.

set	median	mean	stddev	min	max	BER
Control, labelled	59.3255	60.7293	21.8072	29.3255	97.5178	8.75%
Control, unlabelled						10.27%
$k = 20$, **max**	69.4841	82.9839	29.088	35.6346	115.8132	12.59%
$k = 20$, **min**	17.8149	32.365	36.0382	0.0078	93.7374	10.96%
$k = 20$, **base**	61.5314	56.3064	11.0691	35.4286	74.7925	
$k = 80$, **max**	112.6266	108.695	14.7406	34.476	116.3865	10.63%
$k = 80$, **min**	0.0234	5.9211	21.3565	0.0042	96.7166	10.54%
$k = 80$, **base**	62.5621	62.3871	6.5455	27.9629	78.4309	

Table 1: Data information on the $log(|P|)$ of the ellipsoids, and mean BER of each set over all fourteen images tested

Each experiment was conducted several times. For BER analysis, we chose the continuous set of images which had the least visual imperfections (screen tearing, light changes, etc).

4.1 Control Points

The control groups in our experiment showed fairly close BER rates on average. The unlabelled data had a standard deviation (8.83%) approximately 2% higher than that of the labelled data (6.99%). The unlabelled data was much more visually apparent than that of the labelled data, as seen by Figure 4.

As expected, the labelled pairs presented much better visual results than their unlabelled counterparts. There were some visual imperfections within the images generated using labelled

data, mostly due to a smaller set of points to choose from. An easy fix for this is to include more data points, which should occur in a later experiment.

4.2 Experimental Points

As expected, the ellipsoids with a minimized volume performed better than those with a maximized volume. While the significance of this difference was lower than expected, a definitive difference did appear - the maximized volume ellipsoids were less effective, visually and recovery accuracy-wise, than their minimized counterparts.

At $k = 20$, we find that the control ellipsoids have a volume 3.3 times greater than the minimized ellipsoids, and 1.7 times smaller than the maximized ellipsoids. At $k = 80$, the difference is much more intense: 2535 times greater than the minimized ellipsoids, and 1.9 times smaller than the maximized ellipsoids. Thus, we expect to see the most decisive results from $k = 80$.

5 Discussion and Future Work

We find that, among the fourteen images tested, the best BER results came from the labelled pairs. This makes intuitive sense, as the labelled pairs were specifically chosen to minimize the recovery BER. The ellipsoids with a minimized volume tended to perform better overall than those which were maximized, which is interesting and worthy of future experiments.

While performing worse as a whole, on a case-by-case basis, the ellipsoids generated with a minimized volume did sometimes perform better than the unlabelled control data. The ellipsoids with a maximized volume tended to perform worse on a case-by-case basis. At $k = 80$, we found much better results from the experimental data than at $k = 20$, which is expected as the number of points per cluster drops. This is indicative of clustering errors at lower k values.

Most interestingly, the visual impact of images embedded with the experimental data was much less severe than the unlabelled control data. We thus can attempt to increase the α of the experimental data until visually equivalent to the control data, which will decrease the BER of message recovery. That is, we can obtain better results from the minimized ellipsoids than we can with the unlabelled data at equivalent levels of visual interference.

Several images were clear outliers in the data, skewing our average BER slightly, but generally persisted among all of the data sets. That is, some images, such as the first in Figure 3, yield poor results due to features and colors within the image, rather than due to the underlying set of color pairs used in embedding.

5.1 Clustering Algorithms

We see that the standard deviation of the volumes of the minimized ellipsoids is quite high. This suggests that the clustering algorithms currently in use aren't packing points together tight enough – if the points are more confined, we can form a tighter closure around them. To remove the possibility of this being an off-hand error, the minimization of several clusters that had abnormally large volumes were recomputed several times, all to no avail – repeatedly, the volumes were very large in comparison to the others obtained.

The sets of maximized volumes also had a fairly high standard deviation – much more so than the control sets, but not quite as high as the minimized sets. This is an inherent problem of the number of bad points in relation to the number of good points in a cluster. As the number of bad points increases, the chances of picking a cluster such that we can remove all of the bad points drops significantly. Considering the poor results of the maximized points, though, fixing this issue may be unimportant.

An experiment should be conducted in which we hold k to be constant, and utilize various clustering methods to distribute points such that we can find more compact regions. Specifically, we should attempt to find points such that the volume can be minimized uniformly, else remove ellipsoids with volumes higher than the average volume of the entire set.

5.2 Weighting the Objective Function

Among other issues present, we did not assign weights to the terms in the objective function. Presently, the optimization problem we are solving assumes we will not be able to find a perfect separating ellipsoid and thus allows for marginal errors (Section 2) that we attempt to minimize. In the objective functions used in this experiment, we attempt to both minimize the volume and these errors without weighting either above the other (similar can be said for maximizing).

In future experiments, the volume should be weighted more heavily than the error margins. Typically, the log volume is between 0 and 100, whereas $u + v$ is typically in the range of $1,000$ to $10,000$.

References

[1] Eric Wengrowski, Kristin J. Dana, Marco Gruteser and Narayan Mandayam. Reading Between the Pixels: Photographic Steganography for Camera Display Messaging.

[2] W. Yuan, K. Dana, A. Ashok, M. Varga, M. Gruteser, and N. Mandayam. Photographic steganography for visual mimo: A computer vision approach. *IEEE Workshop on the Applications of Computer Vision (WACV)* pages 345-352, 2012.

[3] Sopher, Revan. Detecting Planes in Real-Time for Camera Display Communications

[4] Boyle, Joseph A. Extending Photographic Steganography to Android: Real-time Camera-Display Messaging using a Smartphone Camera

[5] Eric Wengrowski. Pattern Recognition Final Project: A Comparison of Clustering Methods for Differential Metamers.

[6] Boyd, Stephen, and Lieven Vandenberghe. *Convex Optimization.* Cambridge: Cambridge UP, 2009. pages 407-08, 429-30.