Pattern Recognition Final Project: A Comparison of Clustering Methods for Differential Metamers

Eric Wengrowski

ABSTRACT

Traditionally, electronic displays have been used only to display information to human users. Recently, however, new research has emerged that shows how cameras and electronic displays can be used to communicate. A major focus of this research has been methods for cameras and displays to deliver and receive messages that are hidden from human viewers. Previous computer vision research has shown how the mismatch in human-vision sensitivity and camera sensitivity functions can be exploited for hidden message communication [1]. In this work, we will compare how different clustering algorithms effect camera recovery accuracy.

1. INTRODUCTION

Metamers are distinct colors that look the same to humans, because they have the same 3-channel mapping in the human vision system. **Differential Metamers** are pairs of different colors that humans cannot distinguish between when shown in succession, but cameras can. The goal is to find a mapping of differential metamers for every possible (8-bit) RGB value. These pairs of differential metamers are 6-dimensional points consisting of

$$\{R_{base}, G_{base}, B_{base}, R_{delta}, G_{delta}, B_{delta}\}.$$

If we have a set a differential metamers for every RGB triple, we can embed invisible but camera-readable secret messages into any arbitrary image or video. These messages have many diverse applications such as a directed alternative to near-field communication (NFC) networks, interaction with televisions and electronic billboards, indoor localization, and dynamic but invisible fiducial tags. These hidden messages would take advantage of existing devices and infrastructure, and would require no specialized hardware. This process is illustrated in Figure 1.

The nonlinearities in display emittance, camera sensitivity, and our own human color sensitivity make direct computation of these differential metamers difficult. Instead, a data-driven approach is used. But training data is difficult to collect. It requires that *both* the camera and human labels must be assigned. The camera sensitivity tests are trivial to automate. But the human sensitivity tests cannot be automated and take lots of time. For this same reason, results are also difficult to label.

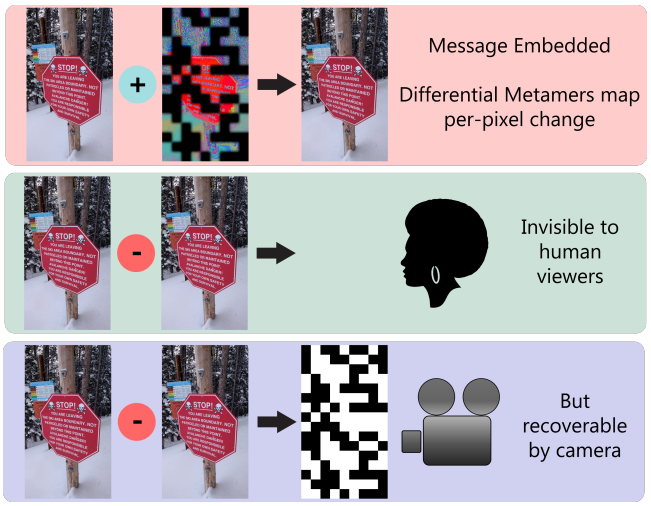

Figure 1: Differential Metamers are color pairs that are optimized to be hidden from human vision, but sensitive to a camera. By modulating small, per-pixel changes in an image, differential metamers can be used to embed hidden messages. [1]

The overall goal of this work is, given an image i_{base}, create embed image $i_embedded$ with per-pixel change d, so: $i_{embedded} = i_{base} + d$. The objective here is to choose d which satisfies the following four axioms:

1. requires no specialized hardware;
2. remains invisible to humans;
3. can be robustly camera-decoded; and
4. satisfies the previous axioms regardless of base-image content.

Previously, there has been no evaluation of the effect of different clustering algorithms in the differential metamers algorithm. In this work, we demonstrate how different clustering algorithms effects the camera-decoded accuracy of dif-

ACM ISBN 978-1-4503-2138-9.
DOI: 10.1145/1235

ferential metamers. Specifically, this work will test implementations of:

- k-Means
- k-Medoids
- Gaussian Mixture Models (GMMs)
- Hierarchical clustering
- Spectral clustering

and measure camera-recovery error as well as run time.

2. RELATED WORK

2.1 Camera-Display Communication

Camera-display messaging is an innovative research area growing in popularity, which embeds a time-varying message invisible for human. Recent work mainly adopt two approaches: 1) high speed light modulation [2, 3], and 2) intensity modulation [4, 5]. Kaleido [2] was developed to prevent videotaping by adding noise to machine-readable channel and retaining the quality of the human-readable channel. The work is based on the concept of "persistence of human vision" that human perceives a blended color when alternating colors are shown at a high frequency, while a camera sees varying colors due to rolling-shutter mechanism and synchronization. A similar approach used for messaging, VRCodes [3] embeds only 1-bit message per frame. Those methods rely on high speed light modulation, and therefore a high speed display is required. Differently, we are seeking a generalized wide bandwidth hidden messaging method without specialized hardware.

Visual MIMO [4] and HiLight [5] employ intensity modulation for message embedding, in which a large intensity change is required for accurate message recovery. However, human vision is generally very sensitive to intensity changes between consequent frames. Even small magnitude intensity modulation is likely to cause flicker and discomfort to humans. Small color gradients were first used to embed watermarks that were difficult to see when far away, but visible up close [6]. Some initial experimentation shows human eye and camera have different spectral color sensitivity. Our method modulates color for messaging, such that it appears unnoticed for human observers but can be distinguished by cameras. A new embedding method we propose to explore texture modulation, which can avoid local intensity change caused by different messages (0,1) embedded into nearby blocks. This approach can preserve higher quality of human observation channel than modulating individual pixels.

Use of Clustering Algorithms

In this paper, different clustering algorithms will be evaluated on the Differential Metamers algorithm. The Differential Metamers approach to camera-display communication uses trained pairs of colors suitable for embedding to generate new color pairs [1]. Similar to Visual MIMO, messages spatially varying 2D barcodes, and these messages are applied to base images with small modulations.

In the Differential Metamers algorithm, k-Medoids is used to initially cluster the training data. However, other clustering algorithms are never evaluated or explored. In this work, we will test how the choice of clustering algorithm effects the camera-recoverability of the result.

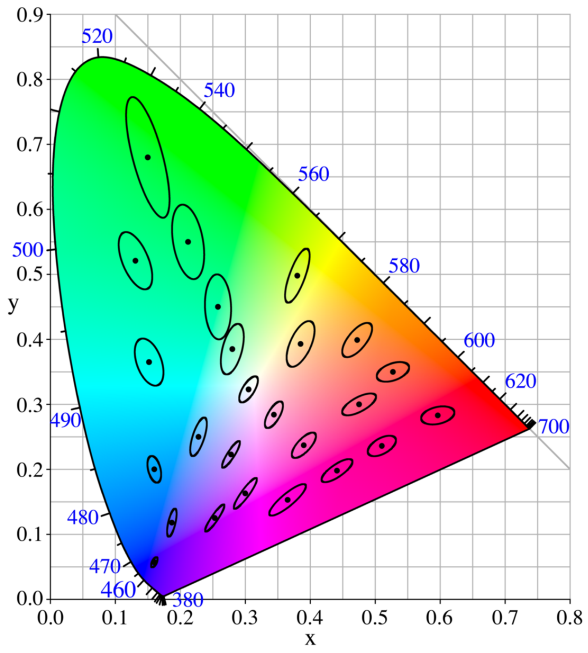

Figure 2: MacAdam ellipses for the CIE xy 1931 colorspace [7, 8]. The area within these scaled-up ellipses represent metamers, or colors which cannot be distinguished.

2.2 Separating Ellipsoids

The use of separating ellipsoids in color space is motivated by two main factors. First, the problem of finding a separating ellipsoid is a convex optimization problem and not affected by local minima. Second, human vision research has showed the utility of ellipsoidal surface fitting for representing color difference thresholds. As early as the 1940's, human vision studies identified and quantified ellipsoidal representations for the problem of understanding human sensitivity to small color differences [9, 7] as illustrated in Figure 2. This ellipsoidal representation was confirmed in numerous studies in early vision literature [10, 11, 12]. Parametric surfaces were used to find discrimination contours and the fitting typically used detection thresholds [13, 14] in order to get just-noticeable-difference JND contours [15]. Our framework greatly simplifies this process because no threshold values are measured. Instead, a separating ellipsoid finds a discrimination boundary between color pairs that are differential metamers and those that are not. Metamer sets [16] are convex hulls, which ellipsoids are well-suited to fit. By extension, we have adopted discriminating ellipsoids to characterize the space of differential metamers.

Using small, selective variations in color to embed information is not new. Rudaz and Hersch adopted small color gradient to watermark spatially varying "microstructures" into images [6]. The objective here was to use color matching to embed watermarks that were difficult to see when far away, but visible up close. On the other hand, our goal is to find pairs of colors where no distinction can be made when viewed sequentially by humans, but the difference can be

robustly detected by a camera.

3. BACKGROUND INFORMATION

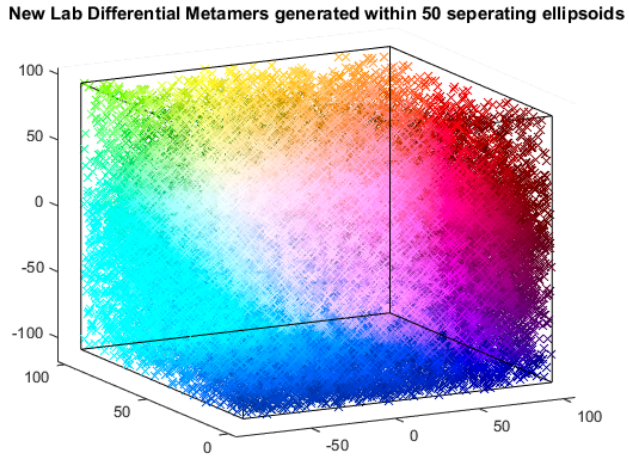

Figure 3: Six-dimensional differential metamers are projected down to *Lab* space. These differential metamers are generated by sampling within the separating ellipsoids. Here, the entire *Lab* space is collectively covered by $k = 30$ ellipsoids.

The space of all possible color pairs is too large to evaluate directly. Even for low-resolution 8-bit *RGB* values, there are 256^6 possible combinations. Instead, we employ a data-driven approach. Given a subset of "good" color pairs, find a superset of "good" color pairs using unsupervised and supervised machine learning techniques. An example superset of Differential Metamers is shown in Figure 3.

Training points labeled as "good" or "bad". These points are in six-dimensional space representing $(i, i + \delta\hat{c})$. "Good" points are defined as ones whose color embedding is invisible to humans, but recoverable by camera with BER (bit error rate) $\leq 5\%$. All other point pairs are negative training data.

A single ellipsoid does not reasonably represent the set of all differential metamers, because color shift is dependent on base color. Therefore we define a separating ellipsoid for each cluster of training data.

The algorithm for finding differential metamers has three main components:

1. Cluster positive training examples into k clusters.
2. For each cluster, find the optimal ellipsoid that separates positive and negative data. An example of this process is shown in Figure 4.
3. Sample within the union of all ellipsoids to find new differential metamers.

To increase the performance of this approach, new differential metamers sampled from within the ellipsoids are labeled and used to retrain the model. This process can be applied to any color space, such as *RGB* or CIE *Lab*.

4. PROPOSED APPROACH

This paper compares the effectiveness of the following clustering algorithms with respect to message recovery:

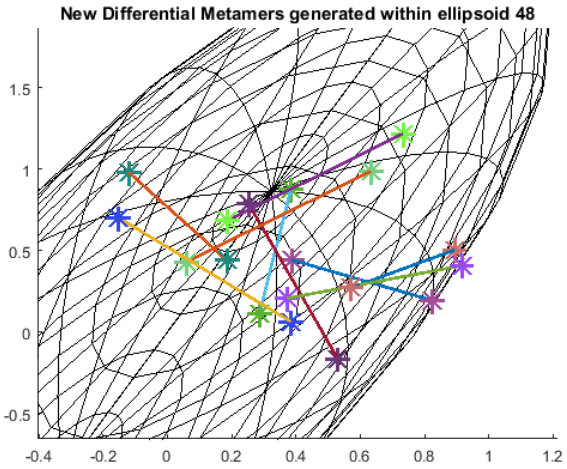

Figure 4: This 6-dimensional separating ellipsoid and subset of differential metamers has been projected down to 3-dimensional space. Each 6-dimensional ellipsoid is trained to contain base colors and respective color shifts that cameras can easily detect, but are invisible to humans. Here, connected nodes represent pairs of base colors and color shifts sampled from within the ellipsoid.

- k-Means
- k-Medoids
- Gaussian Mixture Models (GMMs)
- Hierarchical clustering
- Spectral clustering.

In this section, the specific implementation of each clustering algorithm is discussed.

4.1 k-Means

The k-means++ algorithm was tested [17]. With k-means++, the first cluster center is chosen uniformly at random from the input set. Each subsequent cluster center is chosen randomly from the remaining data points with probability proportional to its distance from the point's closest existing cluster center. A limit of 100 iterations are in place. And this approach is repeated 10 times. The replication with the lowest distortion is selected.

4.2 k-Medoids

This implementation of k-medoids is based on the PAM (partitioning around medoids) algorithm [18]. Seed selection and the swap phase are based on "A simple and fast algorithm for K-medoids clustering." from Park, Hae-Sang, and Chi-Hyuck Jun. [19]. Again, a limit of 100 iterations are in place. And again, this approach is repeated 10 times. The replication with the lowest distortion is selected.

4.3 Gaussian Mixture Models (GMMs)

This implementation of GMMs allows for non-axis-aligned Gaussian distributions. Therefore, these are full Σ Gaussian Mixture Models. The centroids are initiated randomly. The covariance bound is 10^{-6}. Each data point is assigned to

the cluster with the largest posterior. This approach is repeated 10 times. The replication with the lowest distortion is selected.

4.4 Hierarchical clustering

The implementation of Hierarchical clustering evaluated in this paper uses average distance linkage (aka "group average"). The linkage is cut such that there were a maximum of k clusters.

4.5 Spectral clustering

Initially, k-Means is used to organize the eigenvectors. Incidently, this was the same implementation of k-Means discuss earlier in this section.In this implementation of Spectral clustering, we choose $\Sigma = 5$. This approach utilizes an unnormalized Laplacian matrix.

5. EXPERIMENTS

5.1 Experimental Setup

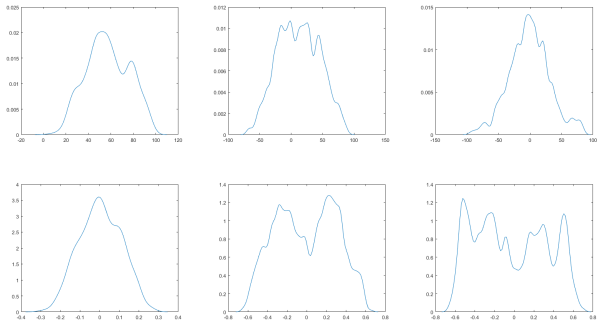

Figure 5: This is the kernel density estimate (KDE) of the training data labeled "good." The first row represents the base colors in CIE *Lab* space, and the bottom row represents the color deltas. The KDE bandwidth of the top row is 4, and the bottom row is 0.025. The KDE reveals that the base colors and the deltas do not come from the same distributions.

For the Differential Metamers generation algorithms was run using each of the clustering algorithms described in the previous section. The clustering and training algorithm is applied to a dataset of 2480 labeled pairs of colors. Within this set there are 1558 "bad" pairs and 922 "good" pairs. Again, "goodness" is defined as pairs of colors where humans can detect no changes, but the camera can recover an embedded message with low error. The dataset can be found at ericwengrowski.com/research/labeled_colors.mat. The Kernel Density estimates of the "good" training data can be seen in Figure 5.

After the ellipsoids are trained and new differential metamers are generated, we measure the accuracy by which a camera is able to recover the embedded message. Using the newly sampled differential metamers, the 14 images shown in Table 1 each have a known message embedded into them. The camera, positioned 1 meter away from the display at a fixed viewing angle, first captures the displayed original image, and then captures the displayed image with a color modulated message. This procedure happens twice for each clustering algorithm under two different illumination conditions. Once where the camera has fixed high-exposure settings, and once again with fixed low-exposure settings.

5.2 Assumptions

Selection of k.

By empirical evaluation, the number of clusters $k = 30$ across all clustering algorithms. The performance of each clustering algorithm would likely be improved if an optimal k was selected for each. However, for simplification of experiments, $k = 30$ because it wasa value that had given good results under a host of experimental conditions.

Camera-Display Scalability.

Generally, individual cameras and displays have unique sensitivity and emittance functions. So the differential metamers trained on one camera-display pair, may produce suboptimal results on another pair. Additionally, camera-display transfer functions may be influenced by white balance, aperture, gain, shutter speed, individual sensor properties.

5.3 Results

Clustering Algorithm	Mean Low Light Error	Low-Light STD	Mean High Light Error	High-Light STD	Run time *(sec)*
k-Means	30.41%	10.65%	24.16%	11.08%	0.0435
k-Medoids	28.97%	10.89%	22.97%	9.92%	0.5813
Gaussian Mixture Models	27.33%	10.83%	22.72%	11.05%	0.0978
Hierarchical clustering	29.37%	11.42%	22.97%	11.64%	0.1299
Spectral clustering	34.52%	11.58%	24.70%	9.91%	0.1387

Figure 6: Camera recovery error for various clustering methods (*lower is better*). Gaussian Mixture Models (GMMs) produce results with the lowest average errors under both illumination conditions. But this advantage is only slight. The large standard deviations indicate that message recovery is highly dependent on the original base image used for embedding.

Under both illumination conditions, Gaussian Mixture Models produce embedding with lower recovery errors on average. But the margin of superiority is generally small. Regardless of method used or illumination condition, the standard deviation hovered around 10% for all methods. This suggests that the recovery error results are largely dependent on the base image used. This result has been verified empirically as well; certain images produce better embedding results. The results are listed in more detail in Table 6.

The run time calculations took place on an Intel 6700K processor with 7% overclock running Matlab 2015b. Although GMMs, were the second fastest clustering algorithm, their run time was more than double k-means. For small

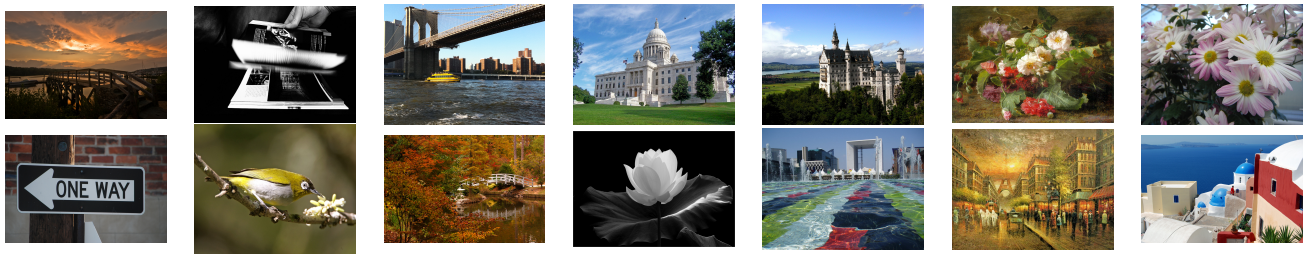

Table 1: Set of 14 images used to evaluate camera recovery accuracy across several clustering algorithms.

training sets like the one use in this paper, this is negligible. But for exponentially larger sets, this may have a serious practical impact on performance.

6. CONCLUSION AND FUTURE WORK

We show that Gaussian Mixture Models outperform other clustering methods when generating differential metamers. Intuitively, GMM clusters are ellipsoidal and work nicely with the separating ellipsoids that are found in a later step. This makes sense since each ellipsoidal GMM cluster will correspond to a single separating ellipsoid.

Since the margin of success was very small, and the standard deviation was large for all evaluated clustering methods, we conclude that the results will mostly depend on the images used for embedding.

However, we can expect our results to change as the classification method changes. Separating ellipsoids are one way of classifying "good" and "bad" color pairs, but other methods can be applied such as deep learning techniques, and kernel SVM. As separating ellipsoids are replaced with new supervised learning techniques, it will be important to reevaluate the effect that each clustering method has on the results. This is left for future work.

This paper does not consider the effects that various clustering algorithms have on human visual perception of embedding. A major component of differential metamers and camera-display messaging in general is hiding images from human perception without flicker or other visual discomforts. In order to truly conclude which clustering algorithms best satisfies the aforementioned axioms of camera-display messaging, a user study needs to be performed to measure human perception.

7. REFERENCES

[1] E. Wengrowski, K. J. Dana, M. Gruteser, and N. Mandayam, "Differential metamers for camera-display messaging," 2016.

[2] L. Zhang, C. Bo, J. Hou, X.-Y. Li, Y. Wang, K. Liu, and Y. Liu, "Kaleido: You can watch it but cannot record it," in *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*, pp. 372–385, ACM, 2015.

[3] G. Woo, A. Lippman, and R. Raskar, "Vrcodes: Unobtrusive and active visual codes for interaction by exploiting rolling shutter," in *Mixed and Augmented Reality (ISMAR), 2012 IEEE International Symposium on*, pp. 59–64, IEEE, 2012.

[4] 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)*, pp. 345–352, 2012.

[5] T. Li, C. An, A. Campbell, and X. Zhou, "Hilight: hiding bits in pixel translucency changes," in *Proceedings of the 1st ACM MobiCom workshop on Visible light communication systems*, pp. 45–50, ACM, 2014.

[6] N. Rudaz and R. D. Hersch, "Protecting identity documents by microstructure color differences," *Journal of Electronic Imaging*, vol. 13, no. 2, pp. 315–323, 2004.

[7] D. L. MacAdam, "Specification of small chromaticity differences," *JOSA*, vol. 33, no. 1, pp. 18–26, 1943.

[8] W. Commons, "Ciexy1931 macadam.png," 2015.

[9] W. Brown and D. MacAdam, "Visual sensitivities to combined chromaticity and luminance differences," *JOSA*, vol. 39, no. 10, pp. 808–823, 1949.

[10] G. Wyszecki and G. Fielder, "New color-matching ellipses," *JOSA*, vol. 61, no. 9, pp. 1135–1152, 1971.

[11] W. R. Brown, "Color discrimination of twelve observers," *JOSA*, vol. 47, no. 2, pp. 137–143, 1957.

[12] H. R. Davidson, "Calculation of color differences from visual sensitivity ellipsoids," *JOSA*, vol. 41, no. 12, pp. 1052–1055, 1951.

[13] A. B. Poirson, B. A. Wandell, D. C. Varner, and D. H. Brainard, "Surface characterizations of color thresholds," *J. Opt. Soc. Am. A*, vol. 7, pp. 783–789, Apr 1990.

[14] G. Wyszecki, V. Stiles, and K. L. Kelly, "Color science: concepts and methods, quantitative data and formulas," *Physics Today*, vol. 21, no. 6, pp. 83–84, 2009.

[15] C. Noorlander and J. J. Koenderink, "Spatial and temporal discrimination ellipsoids in color space," *J. Opt. Soc. Am.*, vol. 73, pp. 1533–1543, Nov 1983.

[16] G. D. Finlayson and P. Morovic, "Metamer sets," *JOSA A*, vol. 22, no. 5, pp. 810–819, 2005.

[17] D. Arthur and S. Vassilvitskii, "k-means++: The advantages of careful seeding," in *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pp. 1027–1035, Society for Industrial and Applied Mathematics, 2007.

[18] L. Kaufman and P. J. Rousseeuw, "Partitioning around medoids (program pam)," *Finding groups in data: an introduction to cluster analysis*, pp. 68–125, 1990.

[19] H.-S. Park and C.-H. Jun, "A simple and fast algorithm for k-medoids clustering," *Expert Systems with Applications*, vol. 36, no. 2, pp. 3336–3341, 2009.