This paper describes objective technical results and analysis. Any subjective views or opinions that might be expressed in the paper do not necessarily represent the views of the U.S. Department of Energy or the United States Government.

SAND2021-14262C

# A Process to Colorize and Assess Visualizations of Noisy X-Ray Computed Tomography Hyperspectral Data of Materials with Similar Spectral Signatures

Joshua Clifford, Emily Kemp, Ben Limpanukorn, and Edward S. Jimenez, *Member, IEEE*

*Abstract*– **Dimension reduction techniques have frequently been used to summarize information from high dimensional hyperspectral data, usually done in effort to classify or visualize the materials contained in the hyperspectral image. The main challenge in applying these techniques to Hyperspectral Computed Tomography (HCT) data is that if the materials in the field of view are of similar composition then it can be difficult for a visualization of the hyperspectral image to differentiate between the materials. We propose novel alternative methods of preprocessing and summarizing HCT data in a single colorized image and novel measures to assess desired qualities in the resultant colored image, such as the contrast between different materials and the consistency of color within the same object. Proposed processes in this work include a new majority-voting method for multi-level thresholding, binary erosion, median filters, PAM clustering for grouping pixels into objects (of homogeneous materials) and mean/median assignment along the spectral dimension for representing the underlying signature, UMAP or GLMs to assign colors, and quantitative coloring assessment with developed measures. Strengths and weaknesses of various combinations of methods are discussed. These results have the potential to create more robust material identification methods from HCT data that has wide use in industrial, medical, and security-based applications for detection and quantification, including visualization methods to assist with rapid human interpretability of these complex hyperspectral signatures.**

## I. INTRODUCTION

THIS paper explores several preprocessing and dimension reduction methods with the goal of representing all channels in a hyperspectral computed tomography (HCT) dataset with one colorized image, portraying objects with maximum smoothness within the objects (each is a homogeneous material) and contrast between objects (to distinguish varying concentrations) to assist with human interpretability and potentially inform material classification.

Past research has visualized hyperspectral data using preprocessing and dimension reduction techniques. For example, Fonville et al. [1] used principal component analysis (PCA), self-organizing maps, and t-distributed stochastic neighbor embedding (t-SNE) to reconstruct mass spectrometry imaging data into a colorized visualization. Similarly, in Rajan et al. [2], PCA and a multi-class support vector machine were used to classify regions within images, and a colorized image was produced as a by-product. Gasser et al. [3] used vertex component analysis to visualize and compare methods for collecting spectroscopy data from plastics. Shi et al. [4] used phasor transformations to both denoise and color hyperspectral fluorescence images. Smoothing splines were used in [5] for noise removal and peak detection in hyperspectral imaging. Numerous studies have developed and explored clustering methods for material classification with hyperspectral data [e.g., 6-8].

Unlike previous work that dealt with hyperspectral data of materials with distinct spectral signatures, our work focuses on the challenge of visualizing materials that have extremely similar material composition or are composed of varying concentrations of the same material. Additionally, this research focuses on applying methods in the context of hyperspectral computed tomography data.

## II. METHODS

### A. Data

The HCT dataset used to develop the methods was generated by PHITS [9], a general purpose Monte Carlo particle transport simulation code. The dataset contains nine cylindrical objects placed in a grid pattern that are imaged in simulation with a patterned anode that contains tungsten, molybdenum, and silver to create a distinct hyperspectral x-ray spectrum from a 225kVp electron beam and detected by an ideal hyperspectral x-ray detector that channelizes the detected photons into 128 energy channels spread out across 300keV [10, 11].

Each cylindrical object is composed of a different mixture of hydrogen peroxide ($H_2O_2$) and water ($H_2O$). The concentration of hydrogen peroxide in the objects ranges from 10% to 90%, with an increment increase of 10% concentration between objects. The layout of the cylindrical objects with their respective $H_2O_2$ concentrations are shown in Figure 1. Ten different sample simulations were generated to allow initial tests of method generalizability, specifically the coloring consistency between different scans of the same objects. In addition, a separate simulation was run with a high photon count (one billion) to provide a reference dataset with less variability in channel intensities.

Manuscript received November 9, 2021.

All authors are with Sandia National Laboratories. Please send any questions or correspondence to Joshua Clifford (email: jmcliff@sandia.gov).

978-1-6654-2113-3/21/$31.00 ©2021 IEEE

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

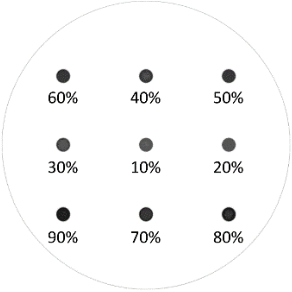

Fig. 1. Reconstructed image of cylindrical objects labeled with respective concentrations of $H_2O_2$ to $H_2O$ (e.g., middle object is 10% $H_2O_2$ and 90% $H_2O$).

### *B. Preprocessing Techniques*

Our approach uses a series of preprocessing methods that isolate objects and spectrally and spatially de-noises the data on a per-object basis. First, negative intensity values in the hyperspectral data were replaced with zeros, as the negative values were an artifact of the reconstruction method used to generate the synthetic data. The next stage of the pipeline was to generate an image mask that isolates the objects from the background. A preliminary mask was first created with a "majority vote" thresholding algorithm that performs multi-level thresholding with the Multi-Otsu algorithm on each channel of the image and then combines the per-channel masks into a single mask. For a pixel in the final mask to be of a certain class, the majority ($\geq$ 50%) of the corresponding pixels in that same location in the per-channel masks must be of that class. Investigations were done to see if this 50% cutoff could be optimized to obtain consistency in the number of pixels in each object between different simulated scans, but this cutoff performed as well as any other. This thresholding method avoids the need for manual selection of a single specific channel to use for multi-level thresholding. This preliminary mask is further refined by applying binary erosion to remove any remaining artifacts and object borders.

Once the objects were isolated by the image mask, we explored several methods to achieve consistency of colors within homogeneous materials. Spatial smoothing was done by convolving a 2D box filter, a 2D Gaussian filter, or 2D median filter over the hyperspectral data for each channel. Smoothing was also done along the spectral dimension by fitting a cubic smoothing spline across the 128 channels. Initial research into coloring techniques using these methods found that a few low energy channels were introducing large amounts of noise and driving a coloring artifact around the edge of each object, which was localized to several channels based on large average differences between edge and middle pixels for each object (channels 2, 3, 6, and 7, which correspond to 2.3-7.0 keV and 11.7-16.4 keV energy bands). Since these channels did not provide useful information (the reconstructed images consisted mostly of noise) and had a negative impact on final coloring, they were dropped from the data. This set of dropped channels may not directly generalize to new data containing scans of different types of materials.

Another method to improve color consistency within an object was to use the partitioning around medoids (PAM) algorithm to group the pixels of each object together. The pixels were clustered based on the intensity per channel augmented with the location of the pixel in the image to encourage clustering based on spatial locality. A smoothing spline was then fit to the aggregated pixels of each group. Alternatively, the median or mean intensity value per channel and object could be used. Both the smoothing spline and the median/mean assignment methods produced an estimate of the spectral signature for each object, either of which was then used by the coloring methods discussed later to assign a color (i.e., RGB value) to each cluster of pixels. These methods improve smoothness and color consistency within each object since all pixels of a cluster were assigned the same spectral signature and, thus, color.

### *C. Coloring Methods*

#### *1) Dimension Reduction*

Based on their use in previous research for similar tasks, numerous methods were tested for representing the high-dimensional hyperspectral signatures in low-dimensional space, such as principal component analysis, nonlinear principal component analysis, self-organizing map, and vertex component analysis. However, the methods presented in this paper's results include t-SNE [12] and Uniform Manifold Approximation and Projection (UMAP) [13]; both are nonlinear manifold learning methods that have been widely applied to visualize high-dimensional data. These two methods were selected for presentation because they had an obvious visual advantage over the others for accomplishing our goals for coloring as well as better performance on our colorization assessment metrics.

To colorize the objects, the 128-channel hyperspectral data were transformed into a three-dimensional space using t-SNE or UMAP, and each dimension was normalized to a scale of [0, 1] for use as a RGB color value. Unless otherwise noted, each plot was produced using ggplot2 [14].

#### *2) Linear Models*

Linear models were explored as an alternative to dimension reduction methods of colorizing hyperspectral data, including logistic regression and polynomial regression. Using logistic regression, rather than directly mapping hyperspectral data to three color channels, the linear model and logit link function were used to represent the concentration of the solution on a pixel-wise basis. Each 128-channel pixel was mapped to a value between 0 and 1 with the link function, which corresponded well with the concentration solution at that pixel. Additionally, channels in the lower and upper frequency ranges that show greater sampling variation were excluded to improve generalizability. The data can then be visualized by mapping the value from the link function to a color gradient. In the examples shown in Fig. 14, matplotlib [15] colormap “inferno” was used to visualize the concentration. A perceptually uniform colormap was chosen to make the ordering more visually interpretable [16].

For the linear model, polynomial terms were utilized to obtain better fits of the nonlinear hyperspectral signatures. The results presented in this paper are those using the quadratic model (where $x$ is the channel and $y$ is the measured absorption, either for a pixel or an object if using median/mean assignment):

$$y = \beta_0 + \beta_1 x + \beta_2 x^2 \tag{1}$$

Using polynomial regression with a quadratic term, the estimated model parameters ($\beta_0$, $\beta_1$, $\beta_2$) were each utilized as a separate input into the RGB bands for coloring. The idea behind this method was to move toward assigning colors based on values representing the shape of the hyperspectral signatures. Since the signatures had a small intensity range (0 - 0.3), the model parameters were small relative to the [0, 1] range used for the RGB bands, which led to low contrast between the objects, so [0, 1] normalization of the model parameters was also attempted to introduce greater contrast.

### *D. Colorization Assessment Metrics*

There are several specific attributes we wish to measure in the colorized image: the variation/inconsistency within each object, the perceptual contrast between the objects, and the average box filter color variation throughout the object (how much variation is in local areas of an object). We created several metrics to estimate each of these specific attributes. All three metrics are based on the notion that humans perceive the RGB color model as an additive model. With this assumption, we can write the variable, $Y'$, which represents the human perception of the red ($R$), green ($G$), and blue ($B$) channels, as:

$$Y' = C_r R + C_g G + C_b B \tag{2}$$

The first three dimensions of the dimension reduction method are used to represent the $R$, $G$, and $B$ channels and are in the range of 0 to 1. The function that creates the RGB value for each pixel from these dimensions rescales the values to be in the range of 0 to 255, thus each $C$ coefficient in Equation (2) is 255. We can now apply the formula for variance on $Y'$ to arrive at:

$$\mathrm{Var}(Y') = \mathrm{Var}(R) + \mathrm{Var}(G) + \mathrm{Var}(B) + 2\mathrm{Cov}(R,G) + 2\mathrm{Cov}(R,B) + 2\mathrm{Cov}(G,B) \tag{3}$$

However, because we aim to utilize $\mathrm{Var}(Y')$ as a metric, we can scale the value and thus neglect the multiplication by $C^2$ on the right hand side of Equation (3). To denote the variance across the RGB channels within object $i$, we can calculate $\mathrm{Var}(Y')_i$, where the vectors $R$, $G$, and $B$ are the red, green, and blue channels for the pixels within that particular object. Now that the variance within each object can be represented, a metric that estimates the variance between objects can be formulated as:

$$\text{Between\_Var} = \mathrm{Var}[\mathrm{Var}(Y')_i] \text{, where } i = 1 \ldots n_{objects} \tag{4}$$

Note that the formulation of Between_Var does not have an upper bound (but are bounded below by 0) like a true metric should. However, it does have a linear relationship with the attributes in the colorized image. Specifically, the smaller Between_Var is, the more similar the variance between objects should be.

While Between_Var is based on a statistical variance approach, we would also like to develop a metric that captures differences in the coloring of the objects, which lead to the most significant perceptual difference between the objects. One straightforward method to measure the difference between two RGB colors is calculating the Euclidean distance like so:

$$\text{Distance} = \sqrt{(R_2 - R_1)^2 + (G_2 - G_1)^2 + (B_2 - B_1)^2} \tag{5}$$

However, calculating this distance for every pair of pixels between two objects, for every pair of objects, would be burdensome. To combat this, we can take the average of the red, green, and blue channels in each object. Equation (6) estimates the color difference between a pair of objects.

$$\text{Est\_dist}_{i,j} = \sqrt{\left(\bar{R}_i - \bar{R}_j\right)^2 + \left(\bar{G}_i - \bar{G}_j\right)^2 + \left(\bar{B}_i - \bar{B}_j\right)^2}\,, \quad where\ (i,j) = ({}_{n_{objects}}C_2) \tag{6}$$

The estimated color difference can be calculated between each pair of objects in the colorized image, and the average of these can be taken to represent the average color difference in an image.

$$\text{Avg\_ColorDiff} = \frac{Avg(est\ dist_{i,j})}{\sqrt{3}}, where\ (i,j) = ({}_{n_{objects}}C_2) \tag{7}$$

The larger Avg_ColorDiff is, the more perceptual color difference between the objects is present in the resulting colorized image. Unlike the previous two metrics, this does have an upper bound ($\sqrt{3} \approx 1.73$). We have normalized this so that (7) is on a scale from 0 to 1.

While both Between_Var and Avg_ColorDiff capture differences between whole objects, oftentimes there is more noise within the objects themselves that we wish to describe with a metric. To develop a metric that can describe this, we need to consider smaller regions within the object and how the color is changing from pixel to pixel within the smaller region. For example, Fig. 2 shows two example objects and respectively two 3x3 regions from the objects, where the first example object has more local color variance (LCV) than the second object.

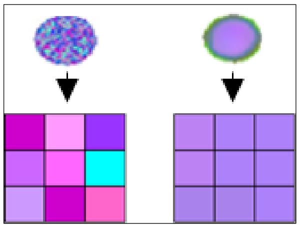

Fig. 2. Example of two objects and a 3x3 region from each object. The object on the left has more LCV in the center than the right object.

Using the same formulation as above for the color variance, Equation (3), we can utilize a sliding box filter to calculate the variance for every 3x3 region in the object. Then, by averaging all these values, we arrive at a metric, Equation (8), that can estimate the amount of LCV within all the objects in the image. Similar to the first metric, there is no upper or lower bound for this metric. However, the smaller this metrics is, the smaller amount of LCV there should be within the objects.

$$\text{Avg\_BoxFilterVariance} = Avg\big(Avg(Y_i')_j\big)\ , \qquad (8)$$
$$where\ i \in 3x3\ window\ in\ the\ object,\ j = 1 \ldots n_{objects}$$

Each of the three developed metrics capture an attribute of interest in the colorized image: Between_Var captures inconsistency (I), Avg_ColorDiff captures contrast (C), and Avg_BoxFilterVariance captures (LCV). Each metric has the relationship that the larger it is, the more of that respective attribute is present in the colorized image.

## III. RESULTS

The median hyperspectral signature for each object that we attempted to colorize can be seen in Fig. 3 below. The absorption profiles are extremely similar since the chemical compositions of the mixture materials are so alike.

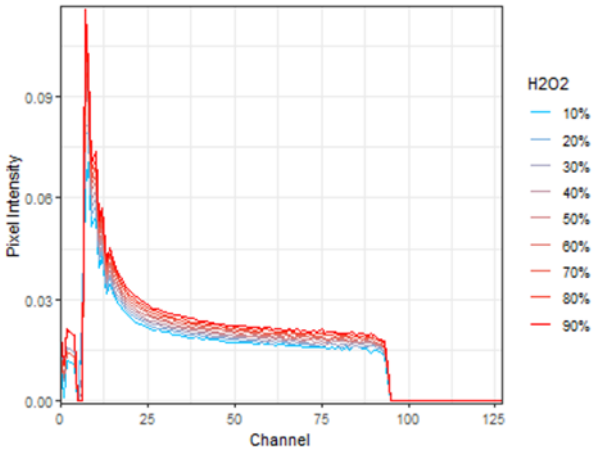

Fig. 3. Median object intensity values over the 128 energy channels.

### A. *Preprocessing*

Before any preprocessing steps, coloring examples on the raw data were produced as a baseline (Fig. 4).

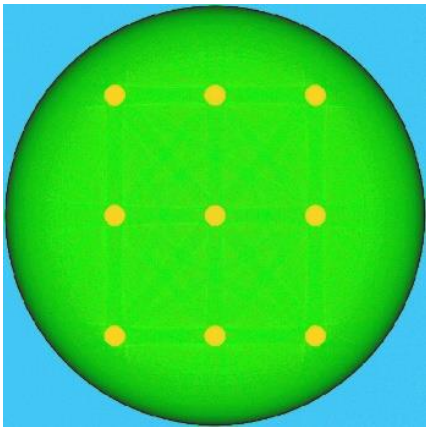

Fig. 4. Example coloring (UMAP) on raw data from sample 1.

First, majority-voting multi-level thresholding was applied. This new method proved to be very successful in removing most of the pixels not associated with objects in the scan (Fig. 5).

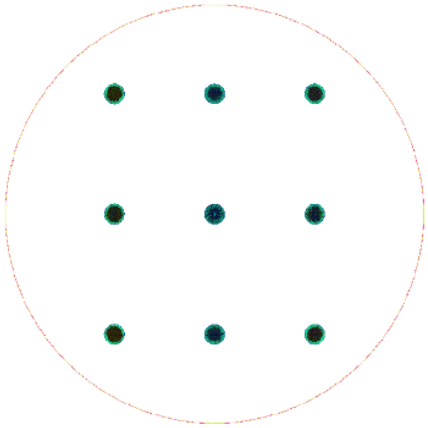

Fig. 5. Example coloring (UMAP, sample 1 data) after majority-voting thresholding.

Binary erosion was applied to remove the remaining artifact encircling the objects due to reconstruction (Fig. 6).

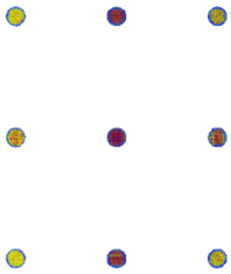

Fig. 6. Example coloring (UMAP, sample 1 data) after thresholding and erosion.

Now that coloring was focused only on pixels within objects, contrast between the objects began to emerge. However, the edges of objects (seen in blue in Fig. 6), were still driving some inconsistency in the coloring of the homogeneous objects. Applying spatial smoothing with a box or Gaussian filter helped smooth the image and increase contrast between objects but also increased the size of the edge artifact (Fig. 7).

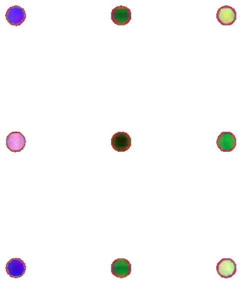

Fig. 7. Example coloring (UMAP, sample 1 data) after thresholding and erosion, smoothed with Gaussian ($s = 1$) filter.

Using the reference data set, we were able to localize the edge pixels into a distinct group by clustering pixels by their intensities over the channels using PAM. We used the result to compare the differences between the edges and centers of each object and discover that there were large average differences for channels 2, 3, 6, and 7. Filtering these channels out of the data was sufficient for the billion photon data to remove the edge artifact and produce good contrast between objects (Fig. 8).

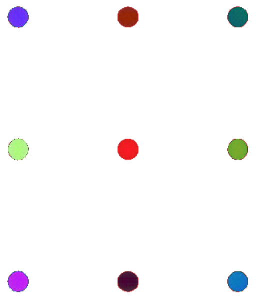

Fig. 8. Example coloring (UMAP, billion photon data) after thresholding, erosion, and channel filtering.

Although this channel filtering method for sample data removed edge artifacts, it was not as effective for providing high levels of contrast between the objects (Fig. 9).

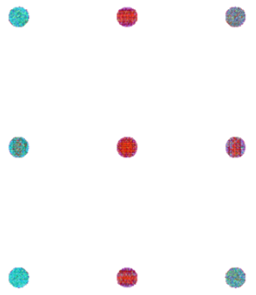

Fig. 9. Example coloring (UMAP, sample 1 data) after thresholding, erosion, and channel filtering.

To better understand why the increased contrast did not carry over to the sample data and how the reference data result could be replicated for the sample data, the differences for each sample pixel and reference pixel over each channel were calculated. We observed that, for most channels (except around the extremes) the differences were normally distributed around zero, suggesting that assigning sample pixels in an object to the mean or median of the object at that channel would provide sample data object contrast similar to the reference data after channel filtering. Augmenting the channel intensities with pixel location data successfully separated objects into distinct clusters using PAM. The median assignment worked as expected (Fig. 10). Mean assignment by object/channel and smoothing splines fit over channels for objects both produced similar results to median assignment.

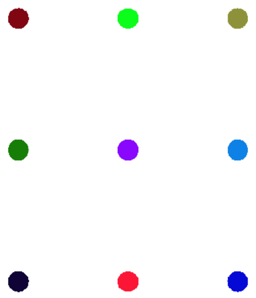

Fig. 10. Example coloring (UMAP, sample 1 data) after thresholding, erosion, channel filtering, and median assignment.

Since median assignment worked effectively for denoising sample data to bring pixels in line with their reference data equivalents, median filters, using [17], were added to testing as an alternative route to median assignment not requiring object detection. Comparisons of multiple radii for the filter suggested a radius size of six would produce the best final coloring scheme, and the method worked fairly well (Fig. 11).

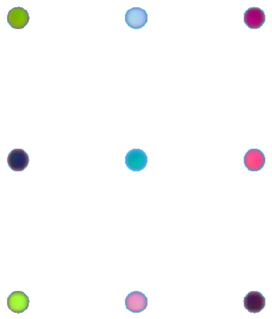

Fig. 11. Example coloring (UMAP, sample 1 data) after thresholding, erosion, median filtering, and channel filtering.

Table I presents metric results, including, contrast (*C*), local color variance (*LCV*), and inconsistency (*I*), for various preprocessing methods tested in this section.

TABLE I. EXAMPLE METRIC RESULTS FOR PREPROCESSING STEPS

| **Preprocessing Steps** | **Image** | **Metrics** |
|---|---|---|
| Thresholding, Erosion | *(Fig. 6)* | $C = 0.1598$<br>$LCV = 0.0409$<br>$I = 0.0009$ |
| Thresholding, Erosion, Spatial smoothing (Gaussian, $s = 1$) | *(Fig. 7)* | $C = 0.2000$<br>$LCV = 0.0233$<br>$I = 0.0050$ |
| Thresholding, Erosion, Median assignment, Channel filtering | *(Fig. 10)* | $C = 0.4908$<br>$LCV = 0.0000$<br>$I = 0.0000$ |
| Thresholding, Erosion, Median filter, Channel filtering | *(Fig. 11)* | $C = 0.2957$<br>$LCV = 0.0118$<br>$I = 0.0025$ |

The metric results validate the visual observations made about each image. Specifically, Gaussian smoothing helps improve contrast between objects and reduce local color variance within objects, but median filters are much more effective at accomplishing these goals. Nevertheless, preprocessing with median assignment was the clear winner, though it necessitates object segmentation and could be more difficult to accomplish with heterogeneous objects or homogeneous objects more closely grouped together in space.

### B. *Coloring with UMAP and t-SNE*

UMAP and t-SNE performed better than other attempted methods for producing smooth and contrasting colorization of objects in images. However, UMAP results have been presented for the examples thus far because it showed additional improvement over t-SNE in the contrast metric. To demonstrate, Fig. 12 below shows a t-SNE visualization produced using the same preprocessing as the UMAP visualization in Fig. 10. In this case, the contrast metric for t-SNE was 0.4276, while the contrast metric for UMAP was 0.4908.

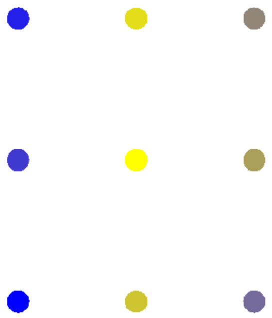

Fig. 12. Example coloring (t-SNE, sample 1 data) after thresholding, erosion, channel filtering, and median assignment.

Testing was completed to see how well UMAP generalized between all ten samples. Using the same random initialization, a manifold for each sample was estimated with UMAP. We discovered that, while the same colors were produced in each sample image, in some samples the object some colors were assigned to changed (Fig. 13). The same issue occurred when the generalization test was run for t-SNE.

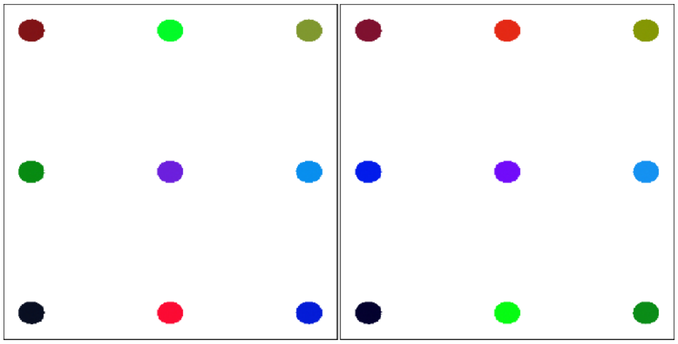

Fig. 13. Example UMAP coloring discrepancy between sample one (left) and sample two (right).

Further analysis into the cause of this discrepancy uncovered it was due to slightly differing numbers of pixels in objects between the different samples, which was slight (usually at most a 4-5 pixel difference for an object) and could not be resolved with adjusting the majority-voting thresholding preprocessing method. This result suggests a larger issue with generalizability of these methods, especially with coloring different images of objects consisting of the same material but having different sizes. However, a workaround was found, where the manifold was learned on data from only one sample and the other samples were dimension reduced by fitting pixels to the existing manifold, which provided consistent coloring between samples.

### C. *Coloring with Linear Models*

While the dimension reduction methods for coloring produced visualizations with contrast, smoothness, and adequate replicability, there are some limitations. First, the coloring is highly dependent on the types of materials in the image and sensitive to size of objects in the image. Second, the interpretability of the object could be improved, such as by assigning not only different colors assigned to each object but, since they are increasing concentrations of a mixture, having the colors be a more meaningful gradient. As such, GLMs were attempted to address these issues.

First, after dropping energy channels under 15 and over 80 due to larger sample variability in objects means between the ten simulations, logistic regression models were trained on either the mean spectrum for each object or on a pixel-by-pixel basis. The output space of the logistic regression models was mapped to sample concentrations and visualized using a gradient color map. This method worked effectively for both the object mean assignment and pixel data (Fig. 14). Additionally, both methods were applied on each of the ten samples and produced extremely similar colorized images, so they proved to generalize well.

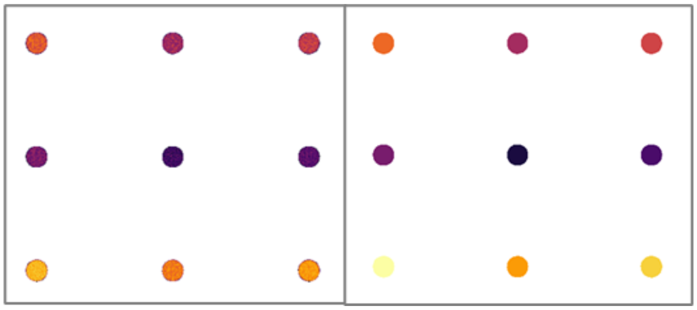

Fig. 1. Example logistic regression coloring on pixels (left) and on mean assignments (right). Both examples use sample 1 data.

Next, we used median assignment and estimated polynomial regression models for the intensity over channels. In this case, we used more of the lower energy channels (only dropping channels 8 and below) to get better fidelity in the model parameters' representation of the hyperspectral signature. We first used the raw parameters for the RGB dimensions, which did not produce much contrast (Fig. 15) because the model parameters were small compared to the [0, 1] ranges used for each color band (model intercepts ranged 0.05-0.065, with linear terms around -0.001, and quadratic terms around 0.0000).

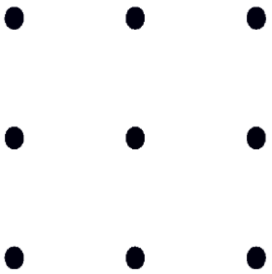

Fig. 2. Example polynomial regression coloring with raw parameters after median assignment (sample 1 data).

To drive contrast between objects using this coloring method, the parameters were each [0, 1] normalized. To some degree, the normalization does remove the advantage of this model being a representation of the real hyperspectral signature; however, if reasonable bounds for model parameters can be determined, then the benefit for visualization contrast from increasing the size of parameters can be retained while avoiding a relative coloring scheme completely dependent upon the types of materials in an image. Additionally, using model parameters for coloring provides flexibility in the coloring scheme (allowing contrast to be increased and color gradients to be tailored to some degree, dependent upon which parameters are normalized and which color bands parameters are assigned to). The example in Fig. 16 normalizes all parameters and uses the linear term, quadratic term, and intercept term for the red, green, and blue color bands, respectively.

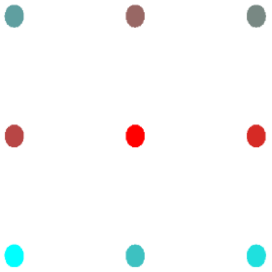

Fig. 3. Example polynomial regression coloring with normalized parameters after median assignment (sample 1 data).

The polynomial regression method for coloring median-assigned pixels works well and generalized to the additional samples. However, initial assessments of model assumptions suggest a potential violation of the linearity assumption that could be solved by adding higher-order polynomial terms to the model. This avenue needs further exploration, especially how parameters could be meaningfully presented in a coloring scheme. Also, while we expect the method to perform well (disregarding the increased computational demand) when used to color individual pixels (based on the logistic regression results), further testing is needed in that regard.

Table II presents metric results for the GLM-colorized images shown above. The logistic regression with mean assignment and polynomial regression with normalized parameters provided the most contrast between objects, with results similar to the t-SNE visualization above. Logistic regression on raw pixels, while noisier, still provided good contrast between objects, with metric results similar to those obtained from median filtering and UMAP. Unsurprisingly, the polynomial regression with raw parameters had very low contrast between objects.

TABLE II. EXAMPLE METRIC RESULTS FOR GLMS

| **Method** | **Image** | **Metrics** |
|---|---|---|
| Logistic regression on pixels | *(Fig. 14)* | $C = 0.2641$<br>$LCV = 0.0444$<br>$I = 0.0004$ |
| Logistic regression on mean assignments | *(Fig. 14)* | $C = 0.3769$<br>$LCV = 0.0000$<br>$I = 0.0000$ |
| Polynomial regression with raw parameters | *(Fig. 15)* | $C = 0.0040$<br>$LCV = 0.0000$<br>$I = 0.0000$ |
| Polynomial regression with normalized parameters | *(Fig. 16)* | $C = 0.4083$<br>$LCV = 0.0000$<br>$I = 0.0000$ |

## IV. DISCUSSION/CONCLUSION

This paper presented several methods for preprocessing images from simulated HCT data and representing their hyperspectral signatures in low-dimensional space to colorize the images. The main visualization goals focused on in the study include smoothness within objects of homogeneous materials and contrast between objects of different concentrations of mixture materials, which were measured with developed metrics that provided quantitative validation of the qualitative assessment of each object. The study results provide a good foundation for continued research into colorization methods for HCT that could enable simple usability and rapid human interpretability of this type of imaging data. These results also have the potential to be utilized as an input for material classification methods.

Several preprocessing steps were identified as indispensable for preparing the HCT data for colorization. The novel majority-voting thresholding technique in combination with erosion was extremely effective in removing pixels not associated with objects, focusing coloring on the objects of interest. These two preprocessing steps removed noise in the colorization due to photon scattering and other artifacts as well as allowed contrast to begin to emerge between the objects. Channel filtering was also uncovered as an effective method for removing remaining object edge artifacts. Further research is needed to develop an appropriate fully automated method for determining the correct channels to remove for objects with different materials than those in the simulated scans for the current study. Additional preprocessing steps, which may or may not be an essential piece of a preprocessing pipeline, nevertheless meaningfully contributed to improved smoothness within and contrast between objects, including use of a median filter for pixels (more than a box or Gaussian filter) and median assignment for objects.

Two alternative methods for translating the pixel or object hyperspectral signatures to a final colorized image were explored, including dimension reduction techniques and GLMs. Among the two best performing methods, UMAP and t-SNE, UMAP provided a clear advantage in contrast between objects. Both methods have generalizability concerns since they are sensitive to object sizes and have results highly dependent upon the objects and materials in an image. However, an alternative method presented, where a pretrained manifold is used to inform subsequent image coloring, showed promise for improving these methods' generalizability but would likely require simulated scans with a vast array of materials to appropriately train for productionized use of such a solution. GLMs provided an alternative path for colorizing images, with the advantage of a more direct connection to the hyperspectral signature of a pixel or object. Logistic regression modeled the concentration of $H_2O_2$ in the mixtures very effectively, even when applied pixel-by-pixel, though the method has a potential limitation of being very specific to the set of material mixtures it is trained on. Polynomial regression parameters provided an effective and flexible framework for colorization of objects, though a higher-order polynomial model would provide a better fit to the hyperspectral signatures and pixel-level models need to be tested in ongoing work. Further analysis of higher-order polynomial regression methods for proper use of model parameters to accomplish HCT colorization in a robust and informative manner is needed for this method to be fully viable for productionization.

Future research will explore the method improvements and tests already mentioned. Colorization metric development will continue, especially for metrics to compare consistency between different scans. Additionally, robustness of the methods will be explored by testing on a variety of additional data, including images with different types of materials (including non-liquids), sizes/arrangements of objects, and concentrations of mixtures. Understanding of characteristics of HCT data will also increase through these tests as well as additional planned testing of effects of varying photon counts in a scan, which would begin to allow characterization of and calibration for drift and degradation of detectors over time. Finally, a real HCT system with the multi-metal patterned anode of molybdenum, silver, and tungsten is under construction, which will further enable HCT data understanding and allow testing of the visualization methods explored in this paper on lab data.

## ACKNOWLEDGMENT

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525. This paper describes objective technical results and analysis. Any subjective views or opinions that might be expressed in the paper do not necessarily represent the views of the U.S. Department of Energy or the United States Government.

## REFERENCES

[1] J. M. Fonville, et al., "Hyperspectral Visualization of Mass Spectrometry Imaging Data," *Anal. Chem.*, vol. 85, no. 3, pp. 1415–1423, 2013.

[2] S. Ranjan, D. R. Nayak, K. S. Kumar, R. Dash, and B. Majhi, "Hyperspectral image classification: A k-means clustering based approach," *4th International Conference on Advanced Computing and Communication Systems (ICACCS)*, 2017, pp. 1-7.

[3] C. Gasser, M. González-Cabrera, M. J. Ayora-Cañada, A. Domínguez-Vidal, and B. Lendl, "Comparing mapping and direct hyperspectral imaging in stand-off Raman spectroscopy for remote material identification," *Journal of Raman Spectroscopy*, vol. 50, no. 7, pp. 1034-1043, 2019.

[4] W. Shi, et al., "Pre-processing visualization of hyperspectral fluorescent data with Spectrally Encoded Enhanced Representations," *Nature Communications*, vol. 11, no.726, pp. 1-15, 2020.

[5] A. Bjorgan, and L. L. Randeberg, "Application of smoothing splines for spectroscopic analysis in hyperspectral images," *SPIE BiOS*, 2019.

[6] I. O. Gallegos, S. Koundinyan, A. N. Suknot, E. S. Jimenez, K. R. Thompson, and R. N. Goodner, "Unsupervised learning methods to perform material identification tasks on spectral computed tomography data," *Proc. SPIE 10763: Radiation Detectors in Medicine, Industry, and National Security XIX*, 2018, doi: 10.1117/12.2326394

[7] H. Yadav, A. Candela, and D. Wettergreen, "A study of unsupervised classification techniques for hyperspectral datasets," *IEEE Geoscience and Remote Sensing Symposium*, 2019.

[8] J. M. Murphy, and M. Maggioni, "Unsupervised clustering and active learning of hyperspectral images with nonlinear diffusion," arXiv:1704.07961v5, 2018.

[9] T. Sato, et al., "Features of Particle and Heavy Ion Transport code System (PHITS) version **3.02,"** *J. Nucl. Sci. Technol.*, vol. 55, pp. 684-690, 2018.

[10] G. Dalton, B. Limpanukorn, E. Kemp, J. Clifford, and E. S. Jimenez, "Monte-Carlo modeling and design of a high-resolution hyperspectral computed tomography system with a multi-material patterned anode for material identification applications," *SPIE Optics+Photonics*, 2021.

[11] E. S. Jimenez, K. R. Thompson, A. Stohn, and R. N. Goodner, "Leveraging multi-channel x-ray detector technology to improve quality metrics for industrial and security applications," *Proc. SPIE 10393: Radiation Detectors in Medicine, Industry, and National Security XVIII*, 2017, doi: 10.1117/12.2275850

[12] L. van der Maaten, and G. Hinton, "Visualizing data using t-SNE," *Journal of Machine Learning Research*, vol. 9, pp. 2579-2605, 2008.

[13] L. McInnes, J. Healy, J. Melville, "UMAP: Uniform Manifold Approximation and Projection for dimension reduction," *arXiv e-prints 1802.03426v3*, 2020.

[14] H. Wickham, ggplot2: Elegant Graphics for Data Analysis. New York, NY: Springer-Verlag, 2016.

[15] J. D. Hunter, "Matplotlib: A 2D graphics environment", *Computing in Science & Engineering*, vol. 9, no. 3, pp. 90-95, 2007.

[16] P. Kovesi, "Good colour maps: How to design them," arXiv:1509.03700, 2015.

[17] S. Perreault, and P. Hebert, "Median filtering in constant time," *IEEE Transactions on Image Processing*, vol. 16, no. 9, pp. 2389-2394, 2007.