An Experiment-based Comparative Analysis of Pigment Classification Algorithms using Hyperspectral Imaging

Dipendra J. Mandal[1], Marius Pedersen[1], Sony George[1], Hilda Deborah[1], and Clotilde Boust[2]

[1]Department of Computer Science, Norwegian University of Science and Technology (NTNU), Norway
[2]Center for Research and Restoration of Museums of France (C2RMF), France

Abstract

Hyperspectral imaging techniques are widely used in cultural heritage for documentation and material analysis. A pigment classification of an artwork is an essential task. Several algorithms have been used for hyperspectral data classification, which are more appropriate than each other, depending on the application domain. However, very few have been applied for pigment classification tasks in the cultural heritage domain. Most of these algorithms work effectively for spectral shape differences and might not perform well for spectra having a difference in magnitude or for spectra that are nearly similar in shape but might belong to two different pigments. In this work, we evaluate the performance of different supervised-based algorithms and some machine learning models for the pigment classification of a mockup using hyperspectral imaging. The result obtained shows the importance of choosing appropriate algorithms for pigment classification.

Keywords – *Hyperspectral Imaging, Pigment Classification, Cultural Heritage, Machine Learning*

1 Introduction

Hyperspectral Imaging (HSI) technology, initially developed and used for remote sensing applications, is also being used more frequently in the Cultural Heritage (CH) domain for

analyzing artwork and has provided great potential in its scientific analysis. In CH, proper pigment classification of artwork materials such as paintings is of essential importance for conservators to precisely analyze an object and its historical value. Generally, reflection, transmission, and absorption of electromagnetic energy by a given material produce a unique spectrum at a given wavelength. The shape of the spectrum is distinctive because every material has a different chemical composition and an inherent physical structure [1]. For pigment classification using HSI, mainly supervised classification algorithms are used; they compare the spectrum within a region of interest with spectral library spectra with a specific tolerance [2, 3].

Many supervised-based classification algorithms exist for HSI, mostly in remote sensing applications, for example, mineral identification [4, 5]. However, few of these algorithms are being adopted directly or with some modification in other application domains such as medical imaging [6, 7], food and agriculture [8, 9], forensics [10]. Moreover, to the best of our knowledge, Only a few have been implemented in the CH domain, especially for pigment classification of artwork such as paintings. HSI acquisition for CH are usually performed under controlled laboratory conditions, where the distance between the camera and the object is relatively small and one has control over illumination types and geometry. In contrast, for remote sensing, HSI data are collected using natural illumination with a more considerable distance between the camera and target, causing temporal illumination variations and atmospheric effects. Due to these differences between two application domains, various classification algorithms adopted in remote sensing cannot be directly adapted or might not work effectively for CH applications. For example, an algorithm insensitive to intensity variation can perform well in remote sensing. However, it might not perform with the same accuracy for CH objects because magnitude measures are essential in CH. Faded or aged pigments [11], pure pigment mixed with different binding mediums [12], mixed

pigments (e.g., pigment mixed in different weight percentages of Lead White [13]), etc. can have variations in magnitude, which is essential to determine for both diagnostic and conservative purposes. Very few of these algorithms have been used for pigment identification of artwork using HSI, and therefore it is necessary to explore and evaluate them. Furthermore, many materials associated with CH lack pure end members, particularly when they undergo weathering [14], aging [15, 16, 17], or restoration processes over time [18]. Therefore, accurately determining the composition of a specific material or differentiating it from other materials within an image can pose challenges, making the task of identifying and mapping materials in HSI more challenging.

Deep learning has recently provided new possibilities by solving more complex questions in many applications [19, 20]. In CH, spectra of the pigments get affected with different types of medium used as binders; spectra might look identical, i.e. might have a small shift in peak or small change in magnitude [12], and under such conditions, most of the supervised algorithms do not perform well for classification. However, distinguishing such conditions might be important for art historians and conservators to select the proper conservation methods. Also, in the case of fading, there might only be a minor change in the magnitude of a spectrum. In medical imaging, Zhi et al. [21] used a Support Vector Machine (SVM) for tongue diagnosis using HSI, where spectra obtained from the surface of the tongue under different conditions have changed mainly in magnitude. Devassy et al. [9] used a One-dimensional Convolutional Neural Network (1D-CNN) to classify strawberries and found that the result was better than supervised algorithms. To the best of our knowledge, using deep learning-based models for pigment classification of artwork is not common practice and, therefore, it will be worthwhile to explore their potential.

This paper presents the comparative experimental analysis of various supervised algorithms and machine learning models for pigment classification on a mockup using HSI in the Visible

Near-Infrared (VNIR) region. The algorithms used are the Spectral Angle Mapper (SAM), Spectral Correlation Mapper (SCM), Spectral Information Divergence (SID), Spectral Similarity Scale (SSS), and the hybrid combinations of SID–SAM and SID–SCM. We also used the Jeffries–Matusita (JM) distance function combined with SAM (JM-SAM). Likewise, some machine learning models used are SVM, Fully Connected Neural Network (FC-NN), and 1D-CNN. The rest of this paper is structured as follows. Section II provides an overview of data processing techniques and algorithms, followed by details about the algorithms used in Section III. Object details, imaging technology, and the experimental framework used are given in Section IV. Section V covers the results with a discussion. Finally, Section VI presents our conclusions, followed by future work.

2 Overview of Algorithms and Processing Techniques

Generally, a spectral matching technique is employed for pigment classification, i.e., finding a spectral similarity between two spectra at any given pixel in an image. The best fit indicates the most significant possibility of being reference material for a given pixel. The distinction between different algorithms used for classification is their ability to consider shapes and magnitude differences between two spectra. This section provides an overview of the classification algorithms employed in various application domains with HSI.

Shivakumar et al. [22] compared the performance of SAM and SCM for classifying nine different classes for remote sensing applications using HSI. There was spectral overlapping between the datasets for some of the classes, and they identified that SCM to be more efficient compared to SAM for the classes with a highly similar spectrum. Similarly, SCM was compared with SAM for mineral analysis [23] and it was found that SCM algorithm has better results due to its wide variation of data from -1 to 1. Qin et al. [24] used SID

methods to identify lesions in citrus using HSI. Devassy et al. [25] experimented to explore the performance of five different algorithms, namely SAM, SCM, ED, SID, and Binary Encoding (BE), for the task of ink classification using HSI. The overall accuracy (average of all inks used) for the SAM algorithm was high compared to all other methods used. None of the methods worked effectively to classify between inks that had nearly similar spectral signatures with only change in magnitude.

For a given two vectors (spectra), Change Vector Analysis (CVA) computes the change in spectral vectors and compares their magnitude with the specified threshold value [26]. It was originally designed for only two spectral dimensions (2 spectral bands); further, using the directional cosine approach, it can be extended to a N-dimensional space [27] and is computed using Equation (1).

$$\alpha_i = \cos^{-1}\left(\frac{t_i - r_i}{\sqrt{\sum_{i=1}^{nb}(t_i - r_i)^2}}\right) \tag{1}$$

where t_i and r_i are the tests and reference image, and nb is the total number of bands with $i = 1, 2, ...nb$. In this method, we will obtain the number of angles α_i equal to the number of bands, which makes the computation complex, details on this explanation and its drawbacks are explained in [28]. Osmar et al. [28] in their study of change detection methods in a tropical environment using HSI, proposes a new approach to calculate the spectral direction of change using the SAM and the SCM method, and for magnitude, they computed the Mahalanobis distance and the Euclidean distance. The best result was obtained using SAM for similarity and ED for magnitude.

Many hybrid approaches to compute the classification of HSI data have shown improved results in many applications. Using a hybrid approach of SAM and SID was found to produce better results than using them alone [29]. Naresh et al. [30] computed the hybrid of SCM and SID (SIDSCM) for the classification of vigna species and compared their result with the hybrid method of SAM and SID. They performed an experiment for various spectral regions and found that for region 400-700 nm it has a better result. Zhang et al. [31] used the hybrid approach by combining Minimum Noise Fraction (MNF) and SAM methods to identify defective tomatoes.

Li et al. [32] proposed a new method called Extended Spectral Angle Mapper (ESAM) for detecting disease in citrus for multi- and hyperspectral datasets. The result was compared with supervised methods, Mahalanobis distance, and unsupervised method; k-means and ESAM were found to have better accuracy (86%) than the other two methods (around 64%). Jeffries-Matusita (JM) [33] are mainly used for the separability criterion and optimal band selection, so only the most distinct bands are selected for the data classification task [34, 35]. The JM method is a pairwise distance measure that can be applied mostly to two class cases. Authors have proposed many extensions of JM [36] to use for multiclass classification. The most common is to take the average JM distance computed for all pairs of classes. Deborah et al. [37] evaluated the performance of four different distance functions named Root Mean Square Error (RMSE), Goodness-of-Fit Coefficient (GFC) [38], Jeffrey divergence, and Levenshtein distance on both synthetic and real hyperspectral datasets to find a suitable distance measure for spectral image processing. They found that for the magnitude change, only RMSE followed by Jeffrey divergence performed in the desired way.

Deborah et al. [39] compared different distance functions for pigment classification tasks on HSI datasets with the presence of spectral noise and variations. Intending to identify the appropriate methods based on suitable selection criteria, they found the Euclidean

distance of a Cumulative Spectrum (ECS) to be the most suitable distance function for spectral data. However, in their study, evaluation of these distance functions on artificially simulated spectra and some real spectra from pigment patches from the Kremer pigment chart [40], these charts are screen printed and usually the pigments are in a water-based binder, which might not be the exact representation of the real spectra obtained from an artwork. Bhattacharyya Distance (BD) which measures the separability between two classes, has been used in remote sensing applications frequently [41, 42], was used to select the number of bands required for efficient classification, and then SAM and Support Vector Machine (SVM) were used for identification of stress symptoms in plants [43]

In recent years, machine learning-based classification methods have been popular and extensively used in many different applications. SVM is one of the machine learning approaches used for classification tasks and has shown efficient results, especially when the training data size is relatively small [44, 5]. Deep learning-based convolutional neural network models can learn spectral features more effectively using deeper layers; in many cases, such methods can give us higher classification accuracy than traditional algorithms. Pouyet et al. [45] used the Deep Neural Network (DNN) and compared the result with SAM for pigment identification and mapping using HSI in the SWIR region and found that the DNN model produced better results than SAM. Devassy et al. [9], in their study of strawberry classification based on sugar content, found that algorithms SID and SAM, which rely on the spectrum's geometry, did not perform well, as the two reference spectrum were nearly identical in shape and a small difference in the magnitude of the NIR region of the spectrum. They also showed that One-Dimensional Convolution Neural Network (1D-CNN) based classification gives better accuracy (96%) compared to SAM (60%) and SID (58%). Table 1 summarizes the list of algorithms used for HSI data processing, their area of study, and details of the classification/network parameters.

Table 1: Summary of algorithms used for HSI datasets with its applications and model parameters; Th:Threshold value, BS: batch size, LR: learning rate, DR: dropout rate, ReLu: rectified linear unit, HL: hidden layer, CL: convolutional layer, FCL: fully connected layer, KS: kernel size

Algorithms	**Application**	**Wavelength**	**Parameters**
ED	Ink classification [25]	400-1000 nm	-
SAM	Pigment classification [46]	400-1000 nm	Th:0.1
	Mineral classification [47]	380-2500 nm	-
	Ink classification [25]	l400-1000 nm	-
	Minerals and land classification [48]	-	-
SCM	Ink classification [25]	400-1000 nm	-
	Pigment identification [3]	370–1100 nm	-
	Pigment mapping [49]	400-2500 nm	Th:0.1
SID	Mineral classification [47]	380-2500 nm	-
	Ink classification [25]	400-1000 nm	-
	Minerals and land classification [48]	-	-
	Crops classification [50]	200-2400 nm	-
SSS	Crops classification [51]	-	-
SIDSAM	Crop classification [29]	400-2500 nm	-
	Mineral classification [47]	380-2500 nm	-
	Dye and pigment based Inkjet prints [52]	400-1000 nm	-
SIDSCM	Plant classification [30]	350-2500 nm	-
	Mineral classification [47]	380-2500 nm	-
JMSAM	Landcover classification [53]	-	-
	Mineral classification [47]	380-2500 nm	-
	Ink classification [25]	l400-1000 nm	-
	Dye and pigment based Inkjet prints [52]	400-2500 nm	-
SVM	Tongue diagnosis [21]	400-1000 nm	-
	Crops classification [54]	-	Polynomial Kernel
FC-NN	Aerial images classification [55]	-	BS:500, LR: 0.05, DR:0.25, ReLU
	Pigment classification [45]	1000-2500 nm	HL: 4, LR : 0.001, Adam, ReLU/Sigmoid
1D-CNN	Soil texture classification [56]	400-1000 nm	CL:4, FCL:2, Softmax
	Classification of strawberry [9]	380-2500 nm	Filters: 8, HL: 2, BS:32, KS: 3

3 Classification Algorithms

In this Section, we describe the algorithms used for our experiment in detail.

3.1 Euclidean Distance (ED)

Classification can be computed by calculating the minimum distance between the spectrum to be classified and the reference spectrum of the class. For a given n-dimensional image spectrum t_i and a reference spectrum r_i, the ED between them is defined using Equation (2), where nb is the number of spectral bands. ED is proportional to the magnitude of the squared subtractive difference vector, but not its shape [57].

$$ED = \sqrt{\sum_{i=1}^{nb} (t_i - r_i)^2} \tag{2}$$

3.2 Spectral Angle Mapper (SAM)

SAM is one of the most popular spectral classification methods used in CH applications due to its easy and rapid approach to mapping spectral similarity. SAM, developed by Boardman (1992) [58], measures the spectral similarity between any two spectra (test and reference). Arccosine angles between the two spectra are calculated by treating them as N-dimensional vectors in space, where N is equal to the number of spectral bands. The angle between two spectra is calculated using Equation (3), where α is the spectral angle in radians, t_i is the image spectrum, r_i is the reference spectrum, and nb is the total number of bands. A smaller angle indicates a more decisive match between the spectra. Kruse et al. [58] describe a simplified representation of the spectral angle mapper algorithm using a

two-dimensional scatter plot for two band image data. Since the SAM algorithm measures an angle between two vectors and the angle does not change with the length of the vectors, i.e., insensitive to the gain. Therefore, this algorithm does not consider magnitude shifts in the spectrum; for details, please refer to Osmar et al. [23].

$$\alpha = \cos^{-1}\left(\frac{\sum_{i=1}^{nb} t_i r_i}{\sqrt{\sum_{i=1}^{nb} {t_i}^2}\sqrt{\sum_{i=1}^{nb} {r_i}^2}}\right) \tag{3}$$

3.3 Spectral Correlation Mapper (SCM)

SCM calculates the Pearson correlation coefficient between two spectra. It standardizes the data, centralizing itself in the mean of the test and reference spectra. By applying arccosine, it can be expressed in angles. This algorithm excludes negative correlation and retains shading effect minimization characteristics similar to SAM, resulting in better classification results [23, 28]. SCM can be computed using Equation (4), where α is the arccosine of the spectral correlation measure in radians, t_i and $\bar{t}_i$ are the image spectrum and its sample mean, similarly r_i and $\bar{r}_i$ are the reference spectrum and its sample mean; and nb is the total number of bands.

$$\alpha = \cos^{-1}\left(\frac{\sum_{i=1}^{nb} (t_i - \bar{t}_i)(r_i - \bar{r}_i)}{\sqrt{\sum_{i=1}^{nb} (t_i - \bar{t}_i)^2 \sum_{i=1}^{nb} (r_i - \bar{r}_i)^2}}\right) \tag{4}$$

3.4 Spectral Information Divergence (SID)

SID measures spectral similarity between the spectrum of test and reference data for each pixel based on the concept of divergence, i.e. measuring probabilistic discrepancy between them. The probability distribution of the test and reference spectra is expressed as Equation (5) and Equation (6), respectively [59].

$$p_i = \frac{t_i}{\sum_{i=1}^{nb} t_i} \tag{5}$$

$$q_i = \frac{r_i}{\sum_{i=1}^{nb} r_i} \tag{6}$$

where, t_i is the image spectrum, r_i is the reference spectrum, and nb is the total number of bands. Using these two probability distributions, SID can be calculated with Equation (7).

$$SID = \sum_{i=1}^{nb} p_i \log\left(\frac{p_i}{q_i}\right) + \sum_{i=1}^{nb} q_i \log\left(\frac{q_i}{p_i}\right) \tag{7}$$

3.5 Spectral Similarity Scale (SSS)

SSS evaluates the shape and magnitude difference between two spectra. Granahan et al. [60, 51] used the SSS to analyze hyperspectral atmospheric correction techniques. This algorithm uses the Euclidean distance metric for magnitude, and correlation for comparing the shape of the spectra. This method combines the calculations of both, giving each an equal weighting [61]. SSS has a scale of minimum of zero and maximum of the square root of two; smaller the value, the higher the similarity between the spectrum i.e. if two spectrum are collinear then its SSS value will be equal to zero. SSS can be computed using

Equation (8).

$$SSS = \sqrt{(d_e)^2 + (\hat{r})^2} \tag{8}$$

Here, d_e is the Euclidean distance between two spectra and is computed using Equation (9) and its value ranges from 0 to 1 due to the factor 1/nb.

$$d_e = \sqrt{\frac{1}{nb} \sum_{i=1}^{nb} (t_i - r_i)^2} \tag{9}$$

Equation (10) computes the value for $\hat{r}$, where r is the correlation coefficient between the two spectra and is computed using Equation (11).

$$\hat{r} = (1 - r^2) \tag{10}$$

$$r^2 = \left(\frac{\sum_{i=1}^{nb} (t_i - \bar{t}_i)(r_i - \bar{r}_i)}{\sqrt{\sum_{i=1}^{nb} (t_i - \bar{t}_i)^2 \sum_{i=1}^{nb} (r_i - \bar{r}_i)^2}} \right)^2 \tag{11}$$

3.6 SIDSAM

As the name suggests, SIDSAM is computed by multiplying SID by taking the tangent of SAM or with the sine function of SAM, i.e., by computing the perpendicular distance between two vectors (test and reference). Both of these measures produce similar results [29].

This hybrid computation makes two similar spectra even more comparable and two dissimilar spectra more distinctive, thus significantly improving the spectral discriminability. SIDSAM can be computed as either of the Equations (12) or (13), where SID and SAM can be computed using Equations (7) and (3)respectively.

$$SIDSAM = SID * \tan(SAM) \tag{12}$$

$$SIDSAM = SID * \sin(SAM) \tag{13}$$

3.7 SIDSCM

Similar to SIDSAM, we also tested the hybrid combination of SIDSCM, computed by multiplying SID by either taking a tangent of SCM or with the sine function of SCM [30]. SIDSCM can be computed as either of Equations (14) or (15), where SID and SCM can be computed using Equations 7 and (4) respectively.

$$SIDSCM = SID * \tan(SCM) \tag{14}$$

$$SIDSCM = SID * \sin(SCM) \tag{15}$$

3.8 Jeffries Matusita-Spectral Angle Mapper (JMSAM)

Similarly to SIDSAM, JMSAM is also a hybrid similarity measure algorithm in which the spectral capabilities of both algorithms are orthogonally projected by using either a tangent or a sine function [53]. A smaller JMSAM value indicates a strong match between

the reference and test spectra. It can be computed using either Equation (16) or (17).

$$JMSAM = JMD * \tan(SAM) \tag{16}$$

$$JMSAM = JMD * \sin(SAM) \tag{17}$$

Here, Jeffries-Matusita distance (JMD) is one of the spectral separability measures commonly used in remote sensing applications and can be computed using Equation (18), where B is the Bhattacharyya distance and is computed using Equation (19) and SAM is computed using Equation (3).

$$JMD = 2\left(1 - e^{-B}\right) \tag{18}$$

$$B = \frac{1}{8}\left(\mu_t - \mu_r\right)^T \left[\frac{\sigma_t + \sigma_r}{2}\right]^{-1} \left(\mu_t - \mu_r\right) + \frac{1}{2} ln \left[\frac{\left|\frac{\sigma_t + \sigma_r}{2}\right|}{\sqrt{|\sigma_t||\sigma_r|}}\right] \tag{19}$$

Here, μ_t and μ_r are the mean of the test and reference spectra, respectively; σ_t and σ_r are the covariance of the test and reference spectra, respectively.

3.9 Support Vector Machine (SVM)

Support vector machine is a supervised classification algorithm used in machine learning and has been used successfully for HSI classification tasks [62, 63, 64]. These are usually used to separate two or more data classes using a hyperplane. Objects to be classified are represented as a vector in an n-dimensional space. Then SVM method draws a hyperplane so that all points of one class are on one side of this hyperplane and points of the other class

are on the other side. Of course, there could be multiple such hyperplanes. SVM tries to find the one that best separates these classes by computing the maximum distance between the data points of these classes closest to the hyperplane, also called support vectors. This method is similar to the Neural Network, but instead of computing the weight and bases of each point, SVM adjusts these parameters by computing it only on the support vectors and determining the decision boundaries for classification.

3.10 Fully Connected Neural Network (FC-NN)

In the FC-NN architecture, all the nodes in one layer are connected to the nodes in the next layer. The data are inputted into the first layer of the neural network, where individual neurons pass the data to a second layer. The second layer of neurons does its task, and so on, until the final layer. Each neuron assigns a weight to its input. Once all the input weights flow out of the neuron, they are summed, and biases are added, which help offset the output. These parameters are tuned by optimization during training, that is, compute the error of classification, also called loss, and then tune the weights and biases over many iterations to minimize this loss. The goal of neural networks is to adjust their weights and biases so that they can produce the desired output when applied to new unseen data. One of the common problems when training the network is overfitting (also called generalization error) of the dataset, i.e., Instead of learning, it memorizes the data. To avoid it, one needs to use regularization, i.e., early stopping with dropout layers and changing the network structure and parameters (weight constraint) [65]. A dropout function added to the network helps to disable the neurons randomly. This forces the network to learn how to make accurate predictions with only randomly left neurons, helping the network to prevent overfitting. For further details, see [66, 67].

3.11 One-dimensional Convolutional Neural Network (1D-CNN)

CNN is one of the most popular neural networks used for various computer vision and machine learning tasks [68, 69, 70]. CNN architecture is built using three main layers: convolutional layer, pooling layer, and fully connected layer. As the name suggests, the convolutional layer performs the linear operation between matrixes, that is, convolution between the input neurons and kernel, generating an output activation map. For 1D-CNN, only 1D convolution is performed, that is, scalar multiplications and additions. In this layer, the number of weights is equal to the size of the kernel and does not depend on the input neuron, as in FC-NN. The feature map generated from this layer is passed through pooling a layer which helps to reduce the dimension of the feature map while maintaining the most important information. This helps to introduce translation invariance and reduces overfitting. A fully connected layer takes the output of the pooling layers, flattens them, and turns them into one long vector that can be an input for the next stage, where it applies weights to predict the correct label, and finally outputs the probabilities for each class using the activation function. Figure 1 shows the architecture of a general convolutional neural network [71].

4 MATERIALS AND METHODS

In this section, we describe the mockup and the HSI acquisition laboratory setup, details on the data post-processing steps, and classification algorithms.

4.1 Test Object

As shown in Figure 2, a pigment mockup was prepared and used in a laboratory environment. We used pigment tubes composed of high-stability pigments and oil, purchased

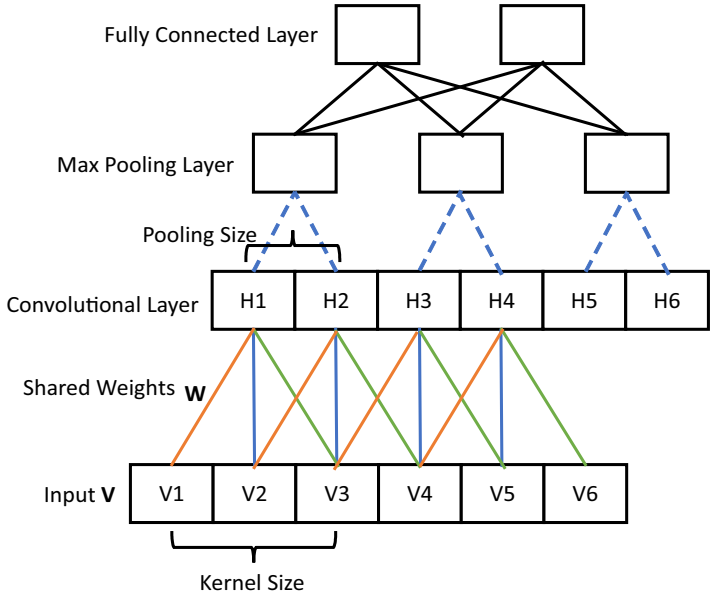


Figure 1: The architecture of a typical CNN consists of a convolutional layer, a max pooling layer, and a fully connected layer.

from Zecchi [72]. The pigments were selected on the basis of the mostly studied in CH research articles, their spectrum characteristics, and consultation with experts. Veridian (V), Cerulean Blue (CB), Green Earth (GE), Yellow Ochre Light (YOL),Burnt Umber (BU), Ultramarine Blue Deep (UBD), Lead White Hue (LWH), Genuine Vermilion (GV), Cobalt Blue Deep (CBD), and Ivory Black (IB) are pigments that are being used in the mockup. The linen canvas used was primed using three layers of white gesso.

4.2 Experimental Setup

Hyperspectral data were obtained in a laboratory environment using the HySpex line scanner VNIR-1800 from Norsk Electro Optikk [73]. The datacube obtained covers a spectral range from 400 to 1000 nm with 186 spectral bands having a spectral resolution of 3.26

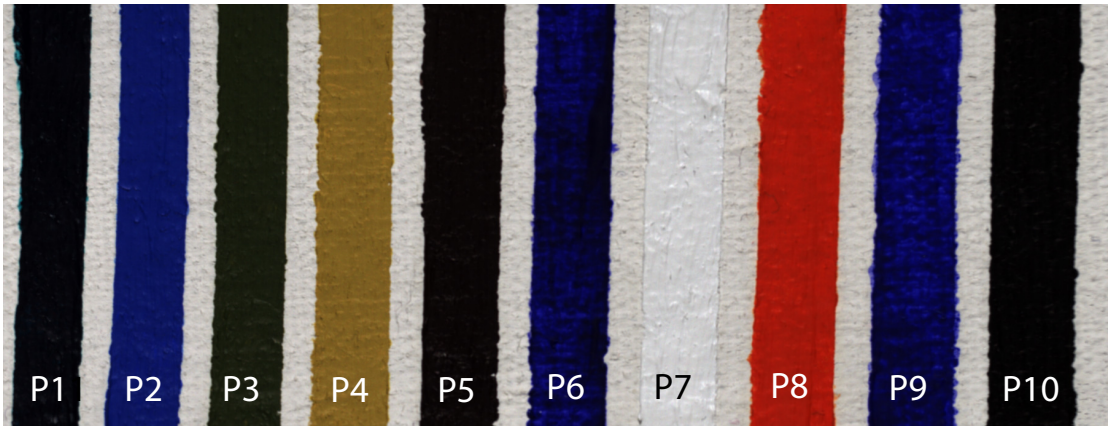


Figure 2: Pigment mockup; P1:Veridian, P2:Cerulean Blue, P3:Green Earth, P4:Yellow Ochre Light, P5:Burnt Umber, P6:Ultramarine Blue Deep, P7:Lead White Hue, P8:Genuine Vermilion, P9:Cobalt Blue Deep and P10:Ivory Black.

nm. In this experiment, a close-range 30 cm lens was used; it captures 1800 spatial pixels across a linear field of view of approximately 86mm. A translation stage setup was used where the pigment mockup was kept lying on a horizontal surface. The standard multistep reference target from Spectralon [74] consisting of four shades of 99, 50, 25, and 12% reflectance values was kept along with the mockup during acquisition. This reference target with a known reflectance factors is used for computing the normalized reflectance at the pixel level.

4.3 Data Processing

The obtained raw hyperspectral data was post-processed for radiometric calibration using the HySpex RAD software, which removes electronics noise, i.e., dark current, and converts the raw images to the sensor absolute radiance values. Illumination correction, i.e., spatial variability in illumination, was performed with the help of the standard reference target. Further data processing steps are different for supervised and ML-based classification and are explained in the following sections.

4.3.1 Data Processing for Supervised Classification

To build a spectral library, a region of interest of approximate size equal to that of the patches (10 × 10 mm) was considered, and the mean spectra from these regions were saved in the library. To evaluate the performance of classification, a confusion matrix was computed. The overall methodology is illustrated using a block diagram in Figure 3. All data processing steps were computed using the open-source software Spectralpython [75].

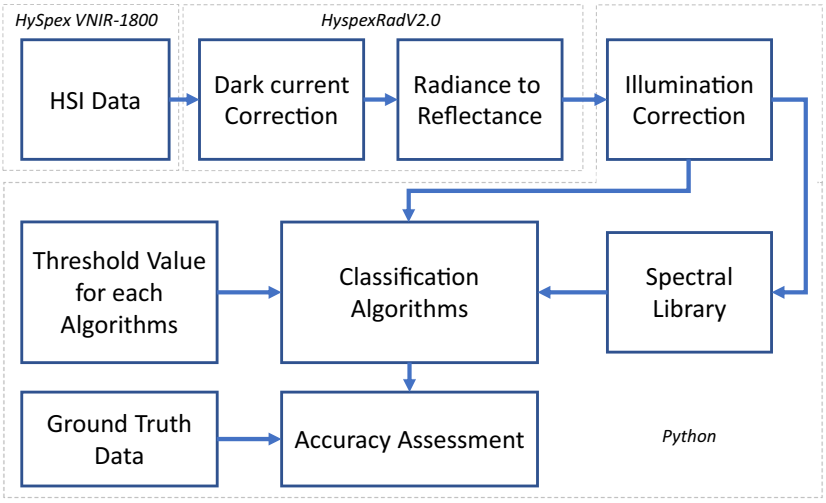


Figure 3: Workflow diagram for data processing

Selecting the appropriate threshold value for classification algorithms is critical as it may vary depending upon the application. For example, Li et al. [32] mentioned the region for selecting the threshold value for SAM to be 0.1 for citrus disease detection analysis because, during the preliminary testing, they found that at a value of 0.15, many false positives result. A similar empirical approach has been followed by Carvalho Júnior et al. [28], and Fung et al. [76]. Thus we also computed the optimal threshold for each of these algorithms through empirical observation. First, we selected a small segment of the HSI dataset of a mockup, as shown in Figure 4. Next, the reference spectrum was extracted

from a mockup's flat region by taking an average of 11 × 11 pixels. Finally, we computed the classification task for all algorithms with different threshold values and evaluated their accuracy using the confusion matrix.

Figure 4: A snippet of a mockup with ten pigments and substrate; Colors are approximated as RGB rendering using spectral python for bands 75, 46, and 19 of HSI datasets.

In CH applications such as pigment classification for a painting, misclassification, i.e., the pigment being classified as the wrong pigment, is even more crucial than a pigment being unclassified. Hence, there should be the minimum error for any given classification algorithm. Therefore, we considered the classification accuracy for pigment classified as correct pigment (P_P), misclassification (MC_), pigment classified as unknown (P_UN_), unknown classified as a pigment (UN_P_), and unknown classified as unknown (UN_UN_). Figure 5 shows the graph for these parameters over accuracy for the SID algorithm, and we can observe that for threshold values between 0.01 to 0.03, the accuracy for pigment classified as pigment and unknown classified as unknown is high. Also, for misclassification value in the range of 0.1 to 0.3, pigments that are classified as unknown is minimum, and unknown classified as unknown is relatively high and constant. A similar conclusion can be drawn by visualizing the classification result shown in Figure 6. An optimal threshold value used for different algorithms in our experiment is mentioned in Table 2 and graph for each of the algorithms is attached in the Appendix B.

4.3.2 Data Processing for ML Classification

The obtained normalized reflectance HSI data needs to process before it is fed to the model; data was labeled for different classes using the label encoder. For our dataset, we used one

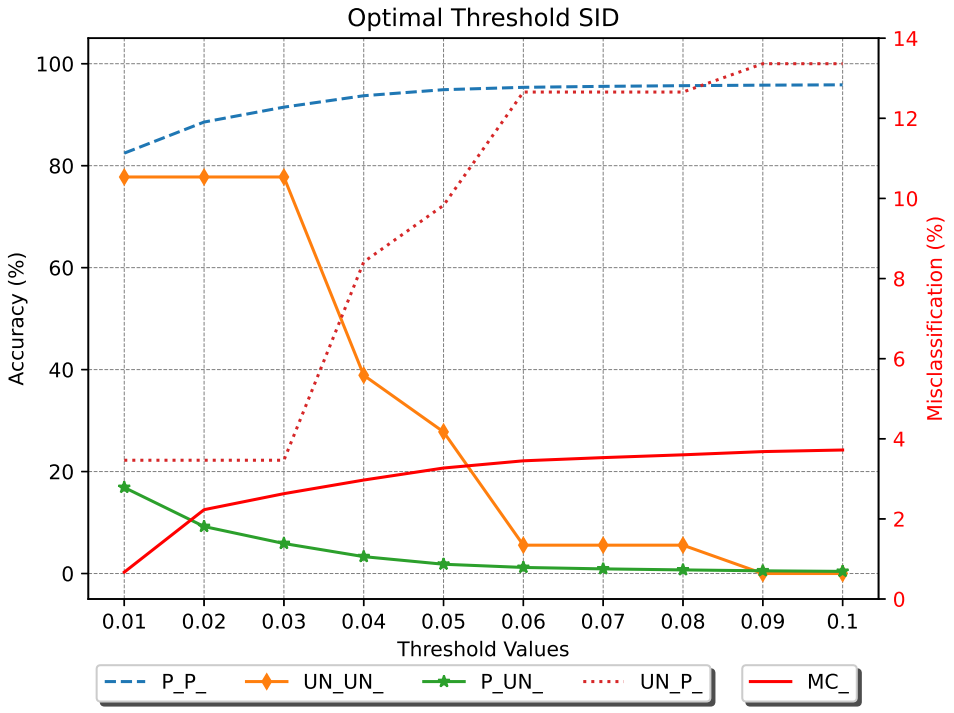


Figure 5: Graph for accuracy of five parameters used to determine the optimal threshold value for the SID algorithm.

Table 2: The selected threshold value for eight different classification algorithms

Algorithms	Threshold Value
ED	0.9
SAM	0.1
SCM	0.8
SID	0.03
SSS	1.1
SIDSAM	0.003
SIDSCM	0.005
JMSAM	0.09

hot encoder, meaning for each class, one value is hot (i.e., the value of 1), and the rest are cold (i.e., the value of 0). We divided the dataset into training and testing, with

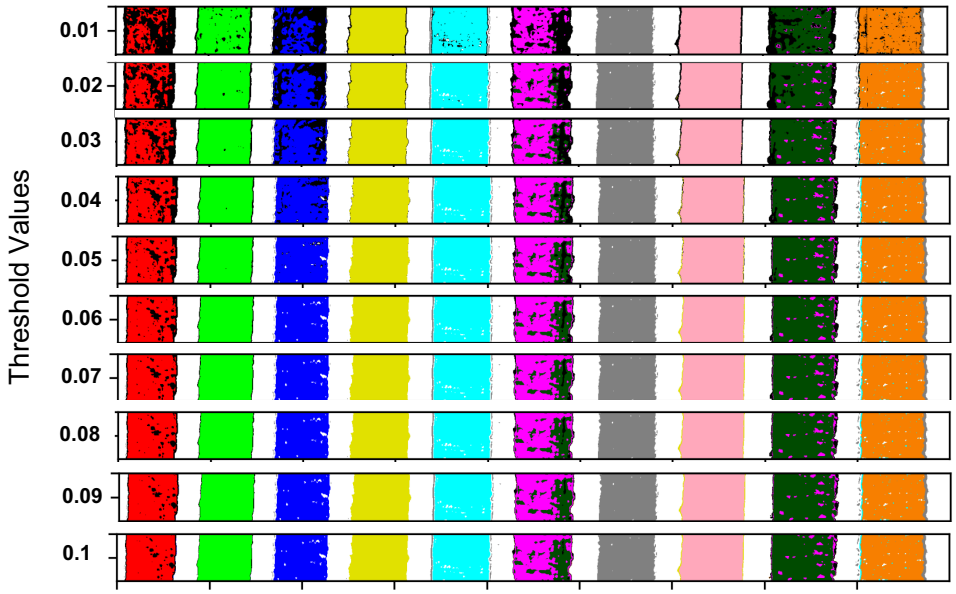


Figure 6: Classification result for ten pigments patches obtained using SID algorithm for a different set of threshold values.

an 80-20 split; Data was further normalized. We then build and implement the model; first training dataset is used to train the model; for neural network weights and biases of neurons are updated with each epoch till we get considerably minimum MSE and higher accuracy. Finally, the test dataset is used to validate the model. A block diagram in Figure 7 illustrates an overall workflow. Training spectra of 10 pigments and a substrate, plotted over a spatial region of approx. 100 x 100 pixels with 186 spectral bands is attached in Appendix A.

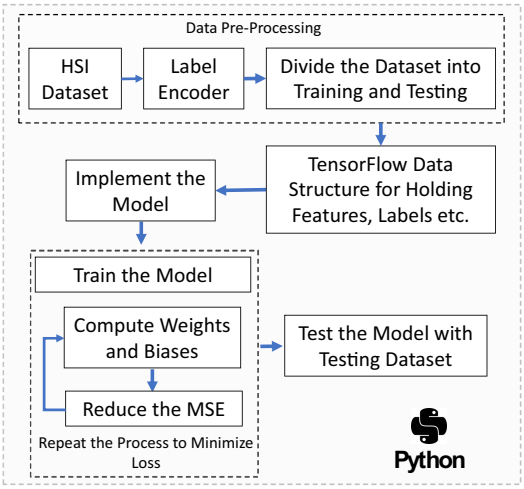


Figure 7: Workflow diagram for ML data processing

SVM model was implemented in Python using the Sklearn library. Among the differences, we tuned our model for three key hyperparameters, namely kernel types, regularization, and gamma, using the Python library called GridSearchCV. This function cross-validates the model to avoid overfitting using k-fold cross-validation and then computes a grid to evaluate the performance of each combination of given hyperparameters. Table 3 shows the details of hyperparameters.

Table 3: SVM key hyperparameters, the range used for tuning, and the optimum value selected for classification; RBF: Gaussian Kernel Radial Basis Function

Hyperparameter	Range Used	Optimum Value Selected
Kernel	'Polynomial', 'RBF', 'Sigmoid', 'Linear'	RBF
Regularization	0.1, 1, 10, 100, 1000	100
Gamma	1, 0.1, 0.01, 0.001	1
k-fold	5	5

For FC-NN, we build a sequential model with three dense layers, the first layer with 32 nodes and hyperbolic tangent (tanh) as activation function followed by banormalizationtion. The

second layer has 16 nodes tanh activation function followed by banormalizationtion and dropout, and the third layer has 11 nodes and a softmax activation function. The activation function introduces the non-linearity into the networks so that the networks can learn the relationship between the input and output. Tangent hyperbolic is a non-linear function with an s-shaped graph with output ranges from -1 to 1. One reason for using the tanh function is that it is zero-centred, which makes the optimization icon process much more manageable. The softmax activation function converts a value vector to a probability distribution and is used in the output layer of multiclass classification. For details on the activation function, please refer [77]. For multiclass classification, the categorical cross-entropy loss function is usually used, and optimizationtion algorithms, which are used to update weights and biases; we used adaptive moment Estimation (Adam), as it is the best among the adaptive optimizers in most of the cases [78, 79]. The network architecture used for our experiment is shown in Figure 8. The model was implemented in Python using Keras, a neural network application programming interface.

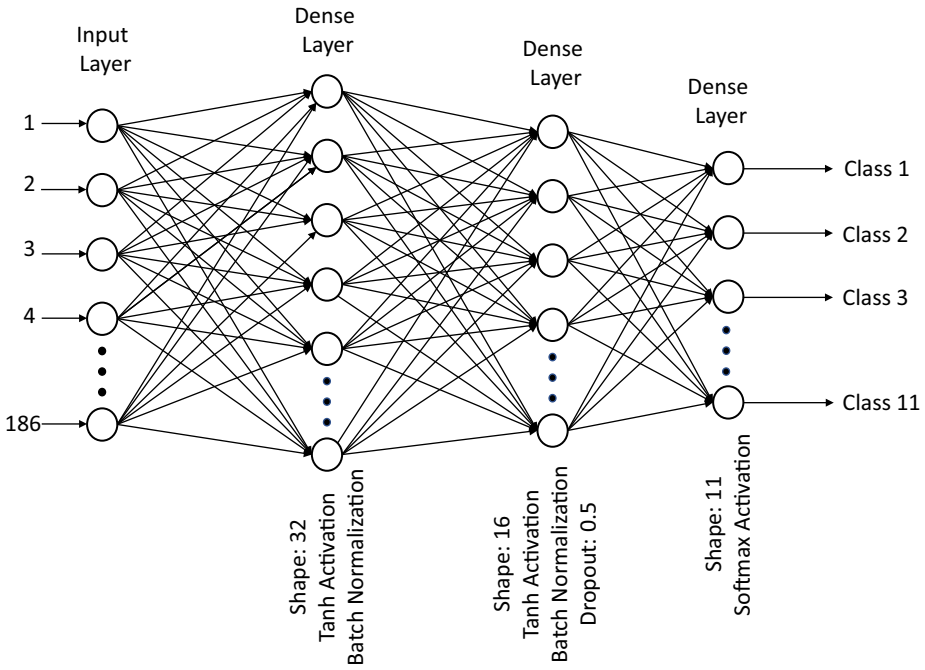


Figure 8: The architecture of the FC-NN classifier, used in our experiment.

The proposed 1D-CNN model was tuned for hyperparameters using KerasTuner [80]. We tuned the model for the number of convolutional layers, their filter size, dropout, dense layer filer size, learning rate and epoch. Figure 9 illustrates the block diagram of the tuned model with its hyperparameter used. We used Adam as an optimizer with a learning rate of 0.001 and categorical cross-entropy as the loss function.

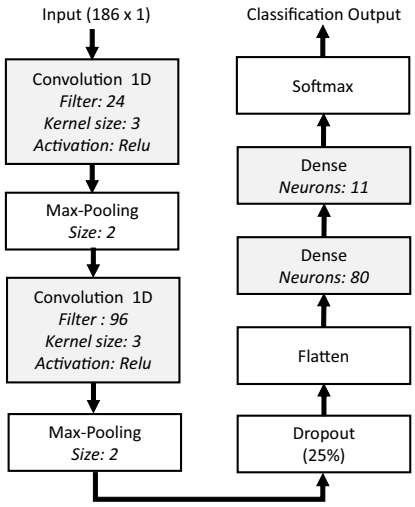


Figure 9: The architecture of tuned 1D-CNN model.

5 RESULT AND DISCUSSION

This section will look in detail at the classified image, the accuracy obtained for each pigment, and the overall accuracy of the algorithms used. Figure 10 shows the classification accuracy of each pigment for the different algorithms. The classification result for each of these algorithms is attached in Appendix D. We can observe that the average accuracy (average of 10 pigments) is high for all three machine learning algorithms. Of these three, FC-NN has the highest accuracy, followed by 1D-CNN and SVM. For the eight supervised algorithms used, SCM and SAM have high accuracy, followed by SID, SIDSAM, SIDSCM, and SSS. ED and JM-SAM have the lowest classification accuracy.

Apart from machine learning algorithms, the other eight algorithms used have difficulty classifying pigment 6 (P6) and pigment 9 (P9). We can see in Figure 11 that spectra for both pigments are similar and have little difference in magnitude. This is a common

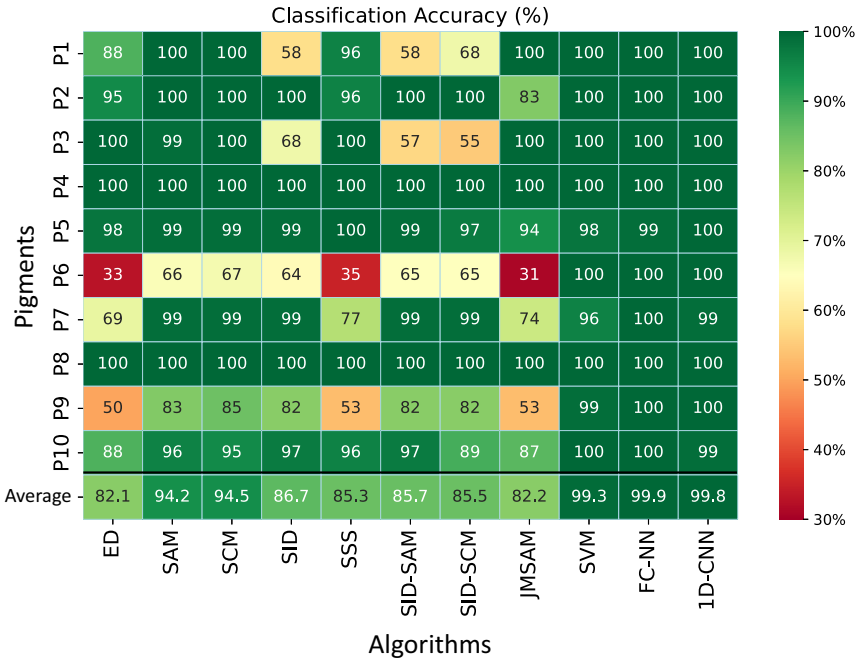


Figure 10: Classification accuracy for each pigment for all 11 algorithms used; average represents the accuracy for an average of 10 pigments for a given algorithm.

issue with supervised-based classification algorithms [9, 22]. In distance-based algorithms, ED, SSS, and JMSAM, the classification accuracy for similar spectra (P6 & P9) are the lowest. We also observed that the classification accuracy is low for these distance-based algorithms, particularly for pigment 7 (White Hue), which has a spectrum similar to the substrate (S), since it is misclassified as substrate, as shown in the confusion matrix in Figure 12. Pigment 10, as shown in Figure 11, has a reflectance value below 0.05 for almost the entire wavelength region (450-1000 nm), and it seems that the low magnitude value has an influence on the classification accuracy for supervised-based algorithms. Spectra for all pigments and substrates used are provided in Appendix C.

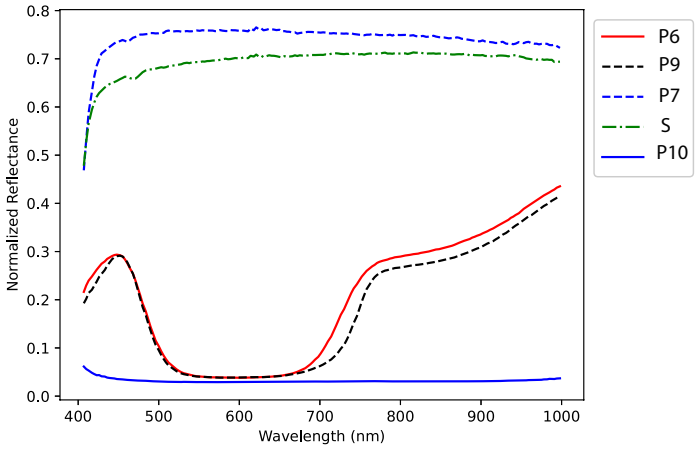
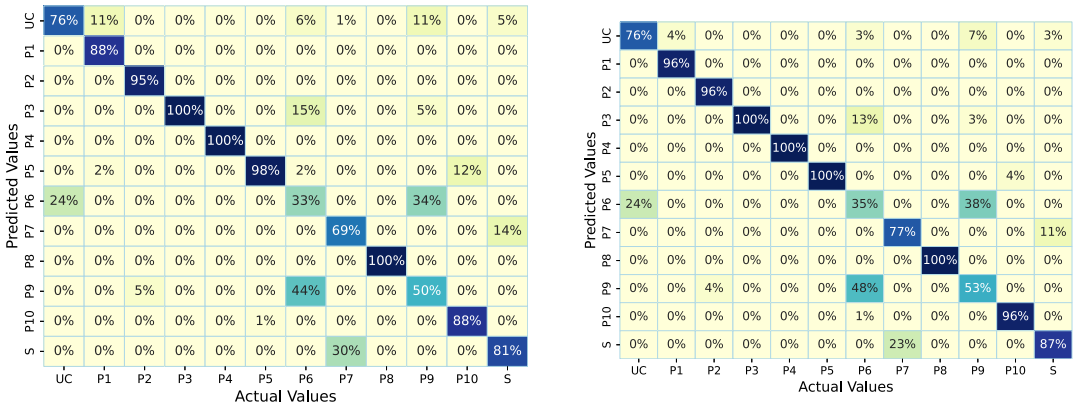
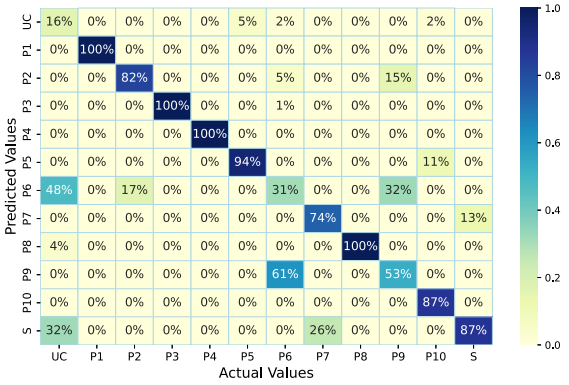


Figure 11: Normalized reflectance spectra for pigment, used as a reference for supervised classification; P6, P7, P9, P10, and S represent pigments 6, 7, 9, 10, and substrate, respectively.



(a) (b)



(c)

Figure 12: Confusion matrix of (a) ED, (b) SSS, and (c) JMSAM

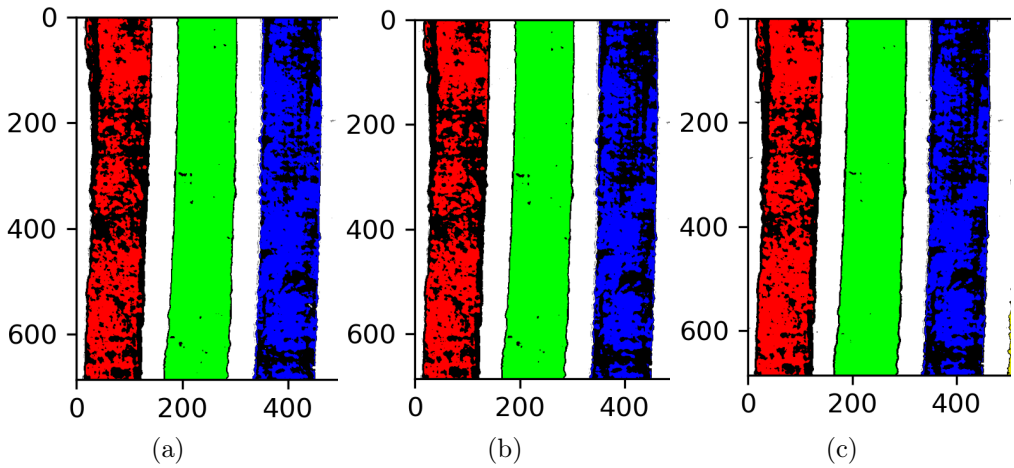


Figure 13: Classification results for pigment P1 (in red), P2 (in green), and P3 (in blue). a, b, and c are obtained using algorithms SID, SIDSAM, and SIDSCM, respectively.

Classification accuracy for algorithm SID and its hybrid combinations (SIDSAM and SIDSCM) are lower for pigments P1 and P3. Figure 13 shows the classification result for pigments P1, P2, and P3 for SID, SIDSAM, and SIDSCM. Black color represents the unclassified pixels, and we can observe that all three algorithms have similar areas that have not been classified for P1 and P3. From the confusion matrix shown in Figure 14, we can see that for P1 and P3, the unclassified (UC) percentage is the second highest value in all three algorithms.

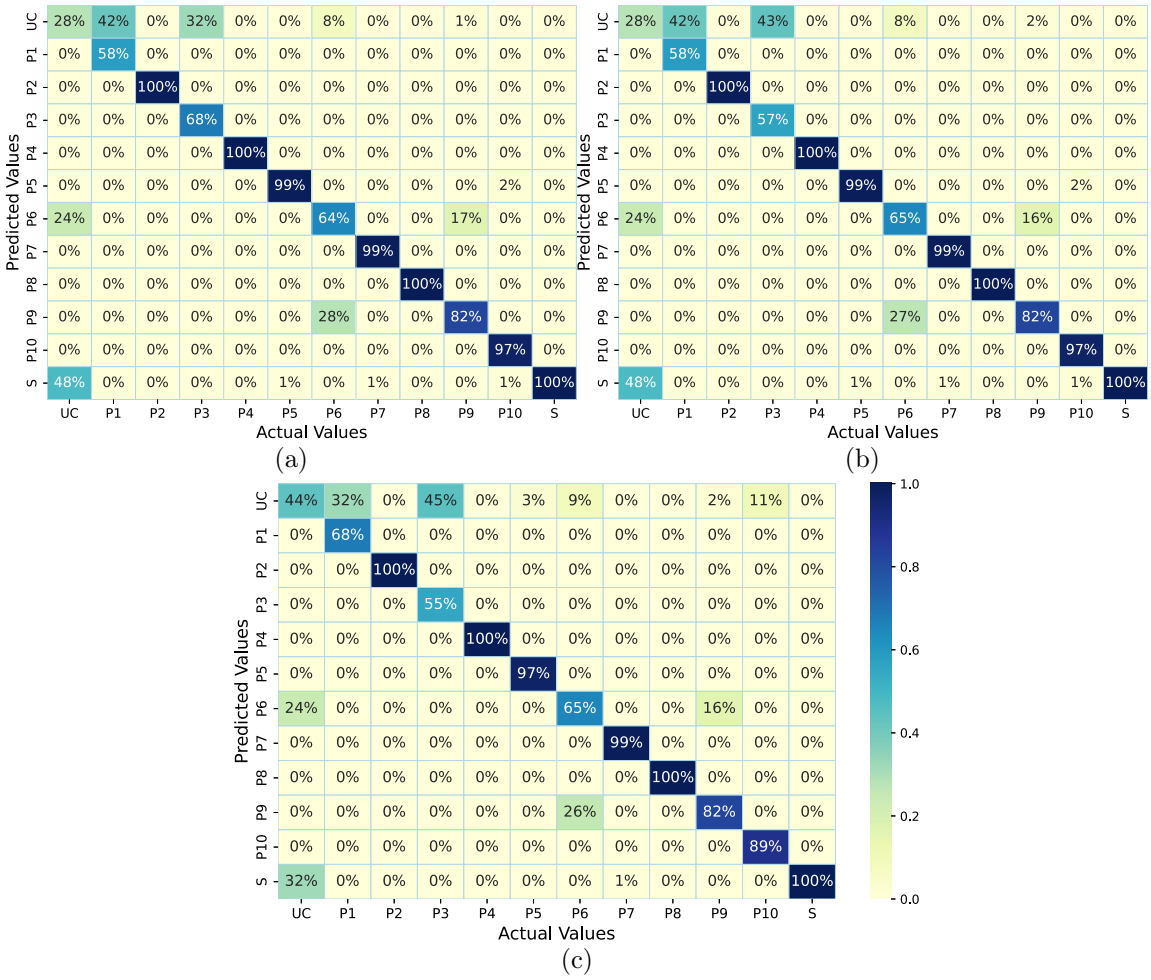


Figure 14: Confusion matrix; (a): SID, (b): SIDSAM, and (c): SIDSCM

Figure 15 shows the spectra for reference, classified pixels, and unclassified pixels for pigments P1, P2, and P3. It can be observed that there is a difference in spectra in the range of 800-1000 nm. The solid red line represents the reference spectrum, whereas red dash lines are spectra for classified pixels, and solid green lines are for unclassified pixels for P1. Similarly, the solid blue line is a reference spectrum for P3, and solid orange and solid black lines are spectra for classified and unclassified pixels, respectively. We also plotted

the range for P2, which is mostly classified. Dashed blue line is a reference spectrum for P2, and solid grey lines are spectra for classified pixels.

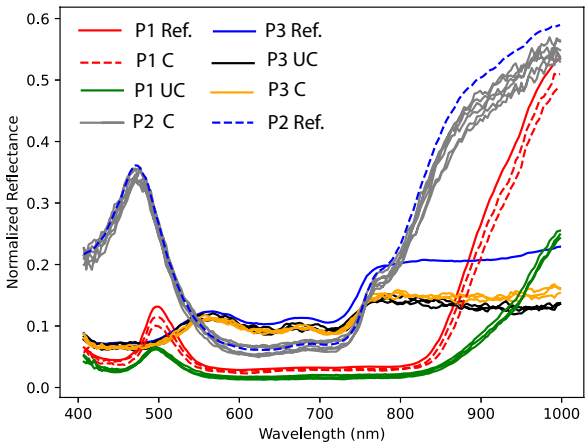


Figure 15: Spectra for pigment P1 and P3; solid red line (P1 Ref.) and solid blue line (P3 Ref.) are reference spectra for P1 and P3, respectively; red dashed line (P1 C) and solid green line (P1 UC) are spectra for classified and unclassified pixels of pigment P1; solid orange line (P3 C) and solid black line (P3 UC) are spectra for classified and unclassified pixels of pigment P3; dashed blue (P2 Ref.) and solid grey (P2 C) are spectra for reference and classified pixels of pigment 2.

The SID algorithm uses a divergence measure to match the reference and target pixels; the smaller the divergence value, the more likely the pixels are similar. We have used a threshold of 0.03, meaning that pixels with a value less than 0.03 will only be classified, and a value greater than the threshold will not be classified. We computed the divergence value for a spectrum of classified and unclassified pixels with a reference spectrum for P1, P2, and P3. The spectra used in the calculation are shown in Figure 16. The computed divergence is shown in Table 4. We can see that spectra that are not classified in the case of P1 and P3 have divergence values greater than a threshold. We can change this value to get more pixels classified, but this will result in higher misclassification and increase the unknown classified as a pigment, as shown in Figure 5.

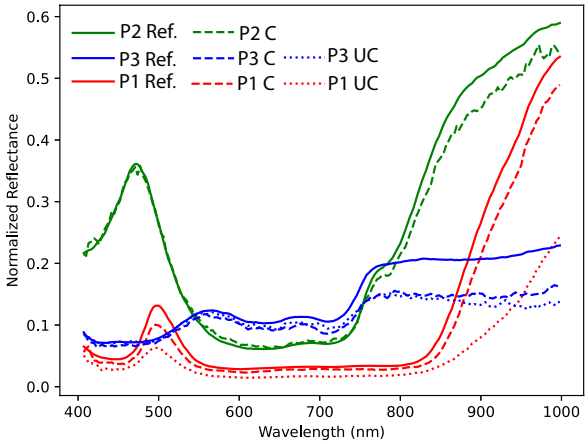


Figure 16: Spectrum of P1, P2, and P3; Ref., C and UC represent reference, classified and unclassified, respectively.

Table 4: SID value computed between a spectrum of reference pixels with that of classified and unclassified pixels for P1, P2, and P3. Remark indicates that either obtained SID value is smaller or greater than a used threshold value of 0.03

Spectra	SID Value	Remark
P1 Ref. & P1 C	0.005	< 0.03
P1 Ref. & P1 UC	0.052	> 0.03
P2 Ref. & P2 C	0.003	< 0.03
P3 Ref. & P3 C	0.014	< 0.03
P3 Ref. & P3 UC	0.034	> 0.03

6 General Discussion

Experimental results show that ML algorithms outperform the supervised-based algorithms used. The limitation of used supervised-based algorithms is that they cannot perform well if pigments have nearly identical spectra (P6 and P9 and also if the magnitude of the spectrum is very low (P10, reflectance factor below 0.05). We found for nearly identical spectrum, SCM is a better measure than the SAM, and this could be because SCM considers value from -1 to 1 whereas the cosine of SAM only varies from 0 to 1. Apart from pigments P1 and P3, we found that the SID's hybrid approach with SAM and SCM has almost similar results

for our dataset. Due to the threshold value selected for classification, the accuracy for P1 and P3 is lower than for other pigments, i.e., in SID for P1 and P3 threshold value should be greater than 0.3 as mentioned in Table 4. The classification accuracy of algorithms based on spectral distance, such as ED, SSS, and JMSAM was the lowest. This could be because these algorithms misclassified in-between white pigment (P7) and substrate (S), which is not the case for other supervised algorithms.

ML-based algorithms need to be trained for which we need a large amount of data. Classification result depends upon how well the model is trained, i.e., how large the training datasets are so that model can learn enough distinct features. For ML-based algorithms to perform well and avoid overfitting of a model, it needs to be tuned for the appropriate value of different hyperparameters, which will take a long computing time. This adds to the cost of computational time and complexity for ML-based algorithms. On the other hand, supervised-based algorithms do not require such a training set and are simple and easy to compute. Therefore, for the pigments with less complex spectra (i.e., having less identical spectrum), supervised-based algorithms such as SCM and SAM might be a good fit for the classification task.

7 CONCLUSION

HSI is a non-invasive imaging technique used for the documentation and analysis of artwork for various tasks, such as pigment classification. It is essential as it assists conservators and curators in precisely analyzing an object and its historical value. In this paper, we evaluated the spectral processing algorithms for pigment classification of a mockup using HSI. We analyzed eight spectral image classification algorithms, i.e., ED, SAM, SCM, SID, SSS, SIDSAM, SIDSCM, JMSAM, and three machine learning-based algorithms, SVM, FC-NN,

1D-CNN for its classification accuracy. In general, machine learning algorithms outperformed the others. Supervised-based algorithms work well for the pigments if their spectra are very distinct in shape from each other. Still, these algorithms have poor performance for pigments having a similar spectrum (nearly identical) or spectrum with just a change in magnitude. However, machine learning-based algorithms can overcome this limitation by extracting the features from each training sample and thus perform better for pigment classification. During our experiment, we trained the network for ten pigments. However, extending the model's scope to include a more extensive range of pigments would be beneficial. Additionally, exploring diverse scenarios, such as mixed and aged pigments, would be beneficial; therefore, one can conduct more comprehensive research in the future. By doing so, we can refine the supervised algorithms and machine learning models mentioned earlier to be more applicable to real-world cases in cultural heritage.

References

[1] G. Shaw and D. Manolakis. Signal processing for hyperspectral image exploitation. *IEEE Signal Processing Magazine*, 19(1):12–16, 2002.

[2] Li Tan and Miao-le Hou. A study on the application of SAM classification algorithm in seal of calligraphy and painting based on hyperspectral technology. In *2016 4th International Workshop on Earth Observation and Remote Sensing Applications (EORSA)*, pages 415–418, 07 2016.

[3] Costas Balas, George Epitropou, Athanasios Tsapras, and Nicos Hadjinicolaou. Hyperspectral imaging and spectral classification for pigment identification and mapping in paintings by el greco and his workshop. *Multimedia Tools and Applications*, 77:9737–9751, 04 2018.

[4] Mahesh Kumar Tripathi and H Govil. Evaluation of AVIRIS-NG hyperspectral images for mineral identification and mapping. *Heliyon*, 5(11):e02931, 2019.

[5] Farid Melgani and Lorenzo Bruzzone. Classification of hyperspectral remote sensing images with support vector machines. *IEEE Transactions on geoscience and remote sensing*, 42(8):1778–1790, 2004.

[6] Ningliang Liu, Yaxiong Guo, Houmin Jiang, and Weisong Yi. Gastric cancer diagnosis using hyperspectral imaging with principal component analysis and spectral angle mapper. *Journal of Biomedical Optics*, 25(6):066005, 2020.

[7] Liu Zhi, David Zhang, Jing-qi Yan, Qing-Li Li, and Qun-lin Tang. Classification of hyperspectral medical tongue images for tongue diagnosis. *Computerized Medical Imaging and Graphics*, 31(8):672–678, 2007.

[8] Bosoon Park, WR Windham, KC Lawrence, and DP Smith. Contaminant classification of poultry hyperspectral imagery using a spectral angle mapper algorithm. *Biosystems Engineering*, 96(3):323–333, 2007.

[9] Binu Melit Devassy and Sony George. Contactless classification of strawberry using hyperspectral imaging. In *CEUR Workshop Proceedings*, 2020.

[10] Deepthi, Binu Melit Devassy, Sony George, Peter Nussbaum, and Tessamma Thomas. Classification of forensic hyperspectral paper data using hybrid spectral similarity algorithms. *Journal of Chemometrics*, 36(1):e3387, 2022.

[11] Jaap van der Weerd, Annelies van Loon, and Jaap J. Boon. Ftir studies of the effects of pigments on the aging of oil. *Studies in Conservation*, 50(1):3–22, 2005.

[12] Antonino Cosentino. Fors spectral database of historical pigments in different binders. *E-conservation Journal*, 2:57–68, 09 2014.

[13] Tiziana Cavaleri, Annamaria Giovagnoli, and Mauro Nervo. Pigments and mixtures identification by visible reflectance spectroscopy. *Procedia Chemistry*, 8:45–54, 12 2013.

[14] David Saunders and Jo Kirby. The effect of relative humidity on artists' pigments. *National Gallery Technical Bulletin*, 25:62–72, 2004.

[15] Shuqiang Lyu, Xueyun Yang, Ning Pan, Miaole Hou, Wangting Wu, Miaomiao Peng, and Xuesheng Zhao. Spectral heat aging model to estimate the age of seals on painting and calligraphy. *Journal of Cultural Heritage*, 46:119–130, 2020.

[16] Jennifer L Mass, Robert Opila, Barbara Buckley, Marine Cotte, Jonathan Church, and Apurva Mehta. The photodegradation of cadmium yellow paints in henri matisse's le bonheur de vivre (1905–1906). *Applied Physics A*, 111(1):59–68, 2013.

[17] Christopher Cuttle. Damage to museum objects due to light exposure. *International Journal of Lighting Research and Technology*, 28(1):1–9, 1996.

[18] Lionel Simonot and Mady Elias. Color change due to surface state modification. *Color Research & Application: Endorsed by Inter-Society Color Council, The Colour Group (Great Britain), Canadian Society for Color, Color Science Association of Japan, Dutch Society for the Study of Color, The Swedish Colour Centre Foundation, Colour Society of Australia, Centre Français de la Couleur*, 28(1):45–49, 2003.

[19] Pramila P. Shinde and Seema Shah. A review of machine learning and deep learning applications. In *2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)*, pages 1–6, 2018.

[20] Lei Cai, Jingyang Gao, and Di Zhao. A review of the application of deep learning in medical image classification and segmentation. *Annals of Translational Medicine*, 8:713–713, 06 2020.

[21] Liu Zhi, David Zhang, Jing-qi Yan, Qing-Li Li, and Qun-lin Tang. Classification of hyperspectral medical tongue images for tongue diagnosis. *Computerized Medical Imaging and Graphics*, 31(8):672–678, 2007.

[22] BR Shivakumar and SV Rajashekararadhya. Performance evaluation of spectral angle mapper and spectral correlation mapper classifiers over multiple remote sensor data. In *2017 Second International Conference on Electrical, Computer and Communication Technologies (ICECCT)*, pages 1–6. IEEE, 2017.

[23] O Abilio De Carvalho and Paulo Roberto Meneses. Spectral correlation mapper (SCM): an improvement on the spectral angle mapper (SAM). In *Summaries of the 9th JPL Airborne*

Earth Science Workshop, JPL Publication 00-18, volume 9. JPL Publication Pasadena, CA, 2000.

[24] Jianwei Qin, Thomas F Burks, Mark A Ritenour, and W Gordon Bonn. Detection of citrus canker using hyperspectral reflectance imaging with spectral information divergence. *Journal of food engineering*, 93(2):183–191, 2009.

[25] Binu Melit Devassy, Sony George, and Jon Y. Hardeberg. Comparison of ink classification capabilities of classic hyperspectral similarity features. In *2019 International Conference on Document Analysis and Recognition Workshops (ICDARW)*, volume 8, pages 25–30, 2019.

[26] William A Malila. Change vector analysis: An approach for detecting forest changes with landsat. In *LARS symposia*, pages 326–335, 1980.

[27] Jin Chen, Peng Gong, Chunyang He, Ruiliang Pu, and Peijun Shi. Land-use/land-cover change detection using improved change-vector analysis. *Photogrammetric Engineering & Remote Sensing*, 69(4):369–379, 2003.

[28] Osmar de Carvalho Júnior, Renato Guimarães, Alan Gillespie, Nilton Silva, and Roberto Gomes. A new approach to change vector analysis using distance and similarity measures. *Remote Sensing*, 3:2473–2493, 11 2011.

[29] Yingzi Du, Chein-I Chang, Hsuan Ren, Chein-Chi Chang, James O Jensen, and Francis M D'Amico. New hyperspectral discrimination measure for spectral characterization. *Optical Engineering*, 43(8):1777–1786, 2004.

[30] M Naresh Kumar, MVR Seshasai, KS Vara Prasad, V Kamala, KV Ramana, RS Dwivedi, and PS Roy. A new hybrid spectral similarity measure for discrimination among vigna species. *International journal of remote sensing*, 32(14):4041–4053, 2011.

[31] Minghua Zhang, Zhihao Qin, Xue Liu, and Susan L Ustin. Detection of stress in tomatoes induced by late blight disease in california, usa, using hyperspectral remote sensing. *International Journal of Applied Earth Observation and Geoinformation*, 4(4):295–310, 2003.

[32] Han Li, Won Suk Lee, Ku Wang, Reza Ehsani, and Chenghai Yang. Extended spectral angle mapping (ESAM)'for citrus greening disease detection using airborne hyperspectral imaging. *Precision Agriculture*, 15(2):162–183, 2014.

[33] M Dabboor, S Howell, M Shokr, and J Yackel. The jeffries–matusita distance for the case of complex wishart distribution as a separability criterion for fully polarimetric SAR data. *International Journal of Remote Sensing*, 35(19):6859–6873, 2014.

[34] Saleem Ullah, Thomas A Groen, Martin Schlerf, Andrew K Skidmore, Willem Nieuwenhuis, and Chaichoke Vaiphasa. Using a genetic algorithm as an optimal band selector in the mid and thermal infrared (2.5–14 μm) to discriminate vegetation species. *Sensors*, 12(7):8755–8769, 2012.

[35] Shilpa Venkataraman, Heidi Bjerke, Kenneth Copenhaver, and John Glaser. Optimal band selection of hyperspectral data for transgenic corn identification. In *MAPPS/ASPRS 2006 Fall Conference*, pages 6–10, 2006.

[36] Lorenzo Bruzzone, Fabio Roli, and Sebastiano B Serpico. An extension of the jeffreys-matusita distance to multiclass cases for feature selection. *IEEE Transactions on Geoscience and Remote Sensing*, 33(6):1318–1321, 1995.

[37] Hilda Deborah, Noël Richard, and Jon Yngve Hardeberg. On the quality evaluation of spectral image processing algorithms. In *2014 Tenth International Conference on Signal-Image Technology and Internet-Based Systems*, pages 133–140. IEEE, 2014.

[38] Javier Romero, Antonio Garcia-Beltrán, and Javier Hernández-Andrés. Linear bases for representation of natural and artificial illuminants. *Journal of The Optical Society of America A-optics Image Science and Vision - J OPT SOC AM A-OPT IMAGE SCI*, 14(5):1007–1014, May 1997.

[39] Hilda Deborah, Noël Richard, and Jon Hardeberg. A comprehensive evaluation of spectral distance functions and metrics for hyperspectral image processing. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 8:3224–3234, 06 2015.

[40] KREMER Pigments, Books & Color Charts. http://www.kremer-pigmente.com. Accessed: 6 October 2022.

[41] Martin Herold, Margaret E Gardner, and Dar A Roberts. Spectral resolution requirements for mapping urban areas. *IEEE Transactions on Geoscience and remote sensing*, 41(9):1907–1919, 2003.

[42] Xin Miao, Peng Gong, Sarah Swope, Ruiliang Pu, and Raymond Carruthers. Detection of yellow starthistle through band selection and feature extraction from hyperspectral imagery. *Photogrammetric Engineering and Remote Sensing*, 73:1005–1015, 09 2007.

[43] Thorsten Mewes, Jonas Franke, and Gunter Menz. Spectral requirements on airborne hyperspectral remote sensing data for wheat disease detection. *Precision Agriculture*, 12(6):795–812, 2011.

[44] JA Gualtieri and S Chettri. Support vector machines for classification of hyperspectral data. In *IGARSS 2000. IEEE 2000 International Geoscience and Remote Sensing Symposium. Taking the Pulse of the Planet: The Role of Remote Sensing in Managing the Environment. Proceedings (Cat. No. 00CH37120)*, volume 2, pages 813–815. IEEE, 2000.

[45] Emeline Pouyet, Tsveta Miteva, Neda Rohani, and Laurence de Viguerie. Artificial intelligence for pigment classification task in the short-wave infrared range. *Sensors*, 21(18), 2021. Article No. 6150.

[46] Dipendra J Mandal, Sony George, Marius Pedersen, and Clotilde Boust. Influence of acquisition parameters on pigment classification using hyperspectral imaging. *Journal of Imaging Science and Technology*, 2021(29):334–346, 2021.

[47] Ramesh Nityanand Adep, Aswin P. Vijayan, Amba Shetty, and H. Ramesh. Performance evaluation of hyperspectral classification algorithms on AVIRIS mineral data. *Perspectives in Science*, 8:722–726, 2016. Recent Trends in Engineering and Material Sciences.

[48] Chein-I Chang. Spectral information divergence for hyperspectral image analysis. In *IEEE 1999 International Geoscience and Remote Sensing Symposium. IGARSS'99 (Cat. No.99CH36293)*, volume 1, pages 509–511 vol.1, 1999.

[49] Hilda Deborah, Sony George, and Jon Yngve Hardeberg. Pigment mapping of the scream (1893) based on hyperspectral imaging. In *International Conference on image and Signal processing*, pages 247–256. Springer, 2014.

[50] Erlei Zhang, Xiangrong Zhang, Shuyuan Yang, and Shuang Wang. Improving hyperspectral image classification using spectral information divergence. *IEEE Geoscience and Remote Sensing Letters*, 11(1):249–253, 2013.

[51] JC Granahan and JN Sweet. An evaluation of atmospheric correction techniques using the spectral similarity scale. In *IGARSS 2001. Scanning the Present and Resolving the Future. Proceedings. IEEE 2001 International Geoscience and Remote Sensing Symposium (Cat. No. 01CH37217)*, volume 5, pages 2022–2024. IEEE, 2001.

[52] Lukáš Krauz, Petr Páta, and Jan Kaiser. Assessing the spectral characteristics of dye-and pigment-based inkjet prints by VNIR hyperspectral imaging. *Sensors*, 22(2):603, 2022.

[53] S Padma and S Sanjeevi. Jeffries Matusita-Spectral Angle Mapper (JM-SAM) spectral matching for species level mapping at Bhitarkanika, Muthupet and Pichavaram mangroves. *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 40(8):1403 – 1411, 2014.

[54] JA Gualtieri and S Chettri. Support vector machines for classification of hyperspectral data. In *IGARSS 2000. IEEE 2000 International Geoscience and Remote Sensing Symposium. Taking the Pulse of the Planet: The Role of Remote Sensing in Managing the Environment. Proceedings (Cat. No. 00CH37120)*, volume 2, pages 813–815. IEEE, 2000.

[55] Manar Ahmed Hamza, Jaber S Alzahrani, Amal Al-Rasheed, Reem Alshahrani, Mohammad Alamgeer, Abdelwahed Motwakel, Ishfaq Yaseen, and Mohamed I Eldesouki. Optimal and fully connected deep neural networks based classification model for unmanned aerial vehicle using hyperspectral remote sensing images. *Canadian Journal of Remote Sensing*, 48(5):681–693, 2022.

[56] Felix Riese and Sina Keller. Soil texture classification with 1D convolutional neural networks based on hyperspectral data. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, IV-2/W5:615–621, 05 2019.

[57] J.N. Sweet. The spectral similarity scale and its application to the classification of hyperspectral remote sensing data. In *IEEE Workshop on Advances in Techniques for Analysis of Remotely Sensed Data, 2003*, pages 92–99, 2003.

[58] F.A. Kruse, A.B. Lefkoff, J.W. Boardman, K.B. Heidebrecht, A.T. Shapiro, P.J. Barloon, and A.F.H. Goetz. The spectral image processing system (sips)—interactive visualization and analysis of imaging spectrometer data. *Remote Sensing of Environment*, 44(2):145–163, 1993. Airbone Imaging Spectrometry.

[59] Chein-I Chang. An information-theoretic approach to spectral variability, similarity, and discrimination for hyperspectral image analysis. *IEEE Transactions on Information Theory*, 46(5):1927–1932, 2000.

[60] James Sweet, James Granahan, and Mary Sharp. An objective standard for hyperspectral image quality. In *Proceedings of AVIRIS Workshop*, 2000.

[61] John P Kerekes, Adam P Cisz, and Rulon E Simmons. A comparative evaluation of spectral quality metrics for hyperspectral imagery. In *Algorithms and Technologies for Multispectral, Hyperspectral, and Ultraspectral Imagery XI*, volume 5806, pages 469–480. SPIE, 2005.

[62] Junping Zhang, Ye Zhang, and Tingxian Zhou. Classification of hyperspectral data using support vector machine. In *Proceedings 2001 International Conference on Image Processing (Cat. No. 01CH37205)*, volume 1, pages 882–885. IEEE, 2001.

[63] F. Melgani and L. Bruzzone. Classification of hyperspectral remote sensing images with support vector machines. *IEEE Transactions on Geoscience and Remote Sensing*, 42(8):1778–1790, 2004.

[64] Sheng Ding and Li Chen. Classification of hyperspectral remote sensing images with support vector machines and particle swarm optimization. In *2009 International Conference on Information Engineering and Computer Science*, pages 1–5, 2009.

[65] H Jabbar and Rafiqul Zaman Khan. Methods to avoid over-fitting and under-fitting in supervised machine learning (comparative study). *Computer Science, Communication and Instrumentation Devices*, 70, 2015.

[66] David Bau, Jun-Yan Zhu, Hendrik Strobelt, Agata Lapedriza, Bolei Zhou, and Antonio Torralba. Understanding the role of individual units in a deep neural network. *Proceedings of the National Academy of Sciences*, 117(48):30071–30078, 2020.

[67] Alexander G Schwing and Raquel Urtasun. Fully connected deep structured networks. *arXiv preprint arXiv:1503.02351*, 2015.

[68] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[69] Shiqi Yu, Sen Jia, and Chunyan Xu. Convolutional neural networks for hyperspectral image classification. *Neurocomputing*, 219:88–98, 2017.

[70] Danfeng Hong, Lianru Gao, Jing Yao, Bing Zhang, Antonio Plaza, and Jocelyn Chanussot. Graph convolutional networks for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 59(7):5966–5978, 2021.

[71] Wei Hu, Yangyu Huang, Wei Li, Fan Zhang, and Hengchao Li. Deep convolutional neural networks for hyperspectral image classification. *Journal of Sensors*, 2015:1–12, 07 2015.

[72] ZECCHI. https://zecchi.it/. Accessed: 10 June 2020.

[73] Norsk Elektro Optikk. http://www.hyspex.no/. Accessed: 20 December 2020.

[74] Spectralon multi-step targets. https://www.labspherestore.com/product-p/aa-006xx-000.htm. Accessed: 11 September 2020.

[75] Welcome to spectral python (SPy). http://www.spectralpython.net. Accessed: 08 August 2020.

[76] Tung Fung and Ellsworth LeDrew. Application of principal components analysis to change detection. *Photogrammetric engineering and remote sensing*, 53(12):1649–1658, 1987.

[77] Tomasz Szandała. Review and comparison of commonly used activation functions for deep neural networks. In *Bio-inspired neurocomputing*, pages 203–224. Springer, 2021.

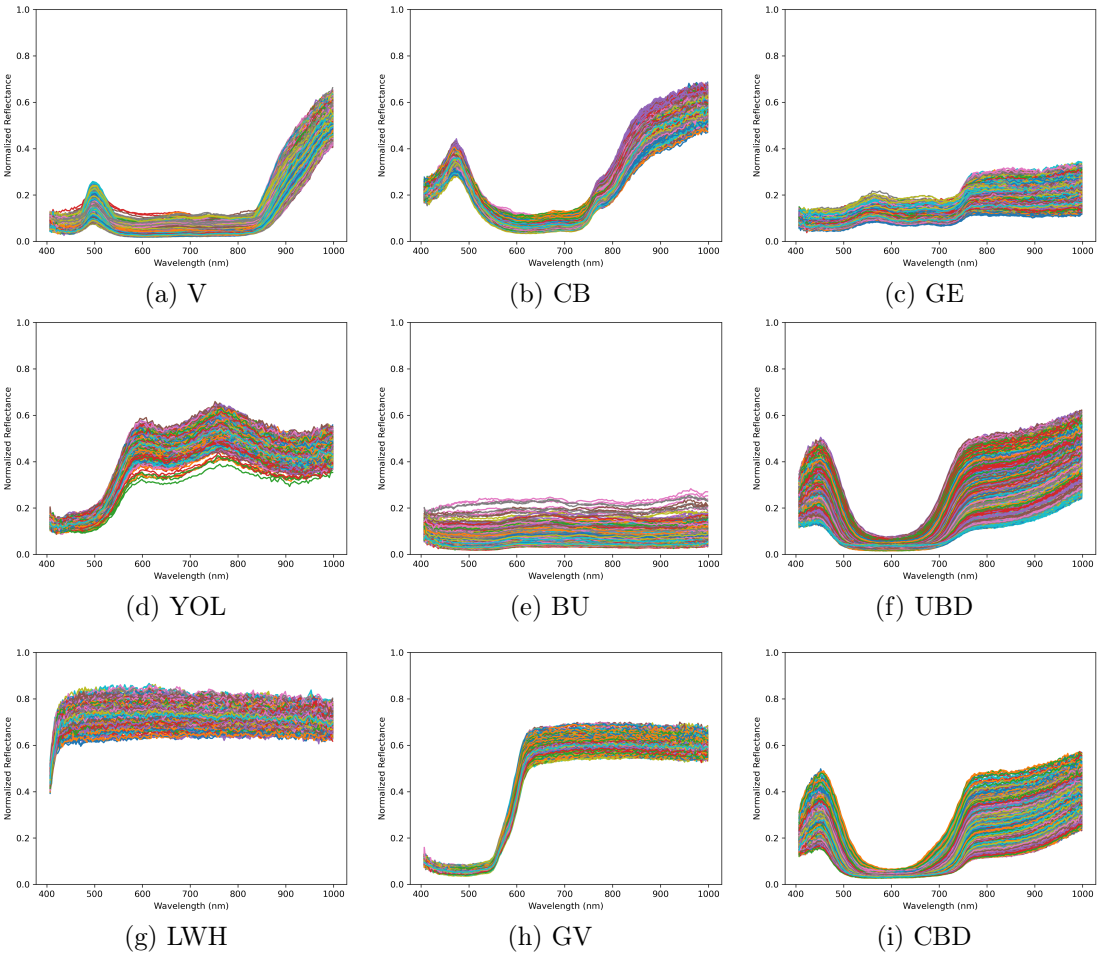
[78] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[79] Muhammad Yaqub, Jinchao Feng, M Sultan Zia, Kaleem Arshid, Kebin Jia, Zaka Ur Rehman, and Atif Mehmood. State-of-the-art CNN optimizer for brain tumor segmentation in magnetic resonance images. *Brain Sciences*, 10(7), 2020. Article No. 427.

[80] Tom O'Malley, Elie Bursztein, James Long, François Chollet, Haifeng Jin, Luca Invernizzi, et al. Kerastuner. https://github.com/keras-team/keras-tuner, 2019. Accessed: 5 December 2022.

Appendix

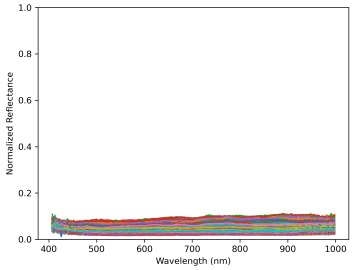
A Reflectance spectra of 10 pigments and substrate used to train SVM, FC-NN and 1D-CNN.



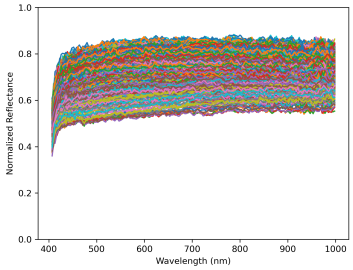
(a) V (b) CB (c) GE

(d) YOL (e) BU (f) UBD

(g) LWH (h) GV (i) CBD



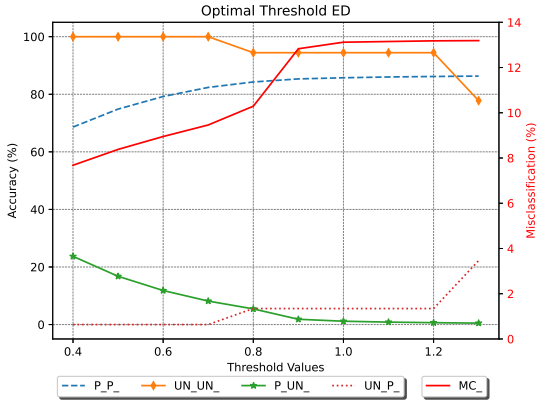
(j) IB



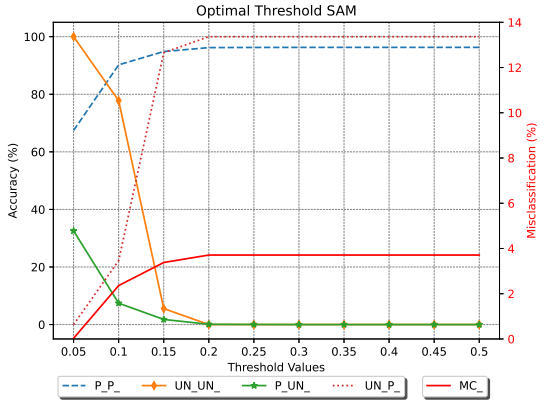
(k) (S)

Figure A.1: Training spectra of 10 pigments and a substrate, plotted over spatial region of approximately 100×100 pixels with 186 spectral bands; Viridian (V), Cerulean Blue (CB), Green Earth (GE), Yellow Ochre Light (YOL), Burnt Umber (BU), Ultramarine Blue Deep (UBD), Lead White Hue (LWH), Genuine Vermilion (GV), Cobalt Blue Deep (CBD), Ivory Black (IB), and Substrate (S).

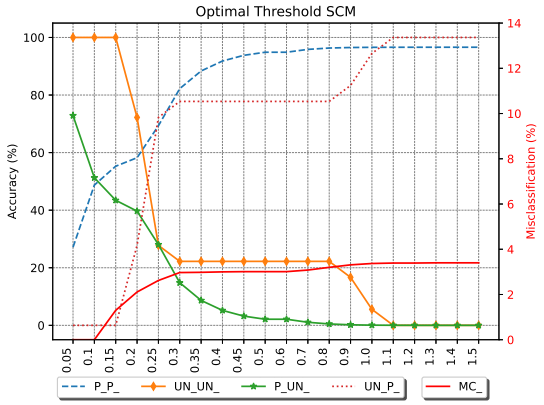
B Graph used for determining the optimal threshold value for different algorithms



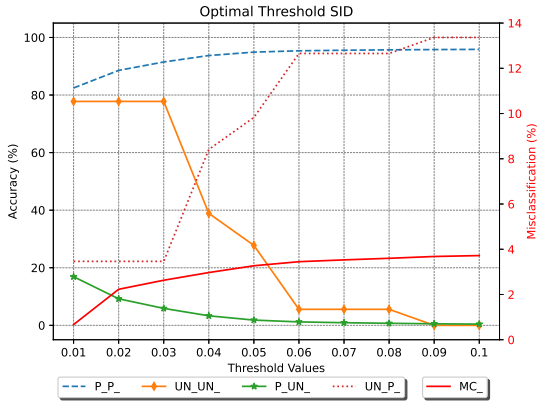
(a) ED



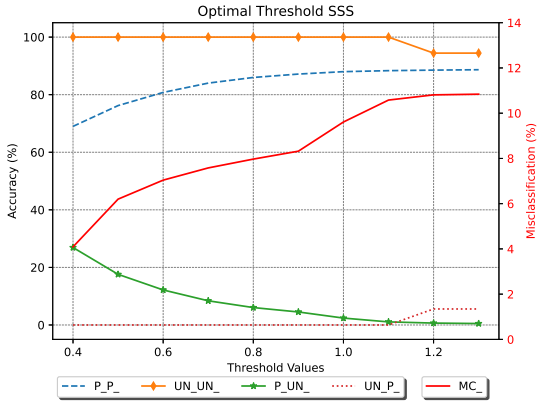
(b) SAM



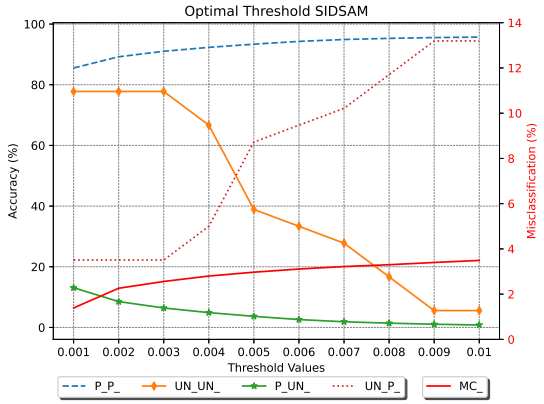
(c) SCM



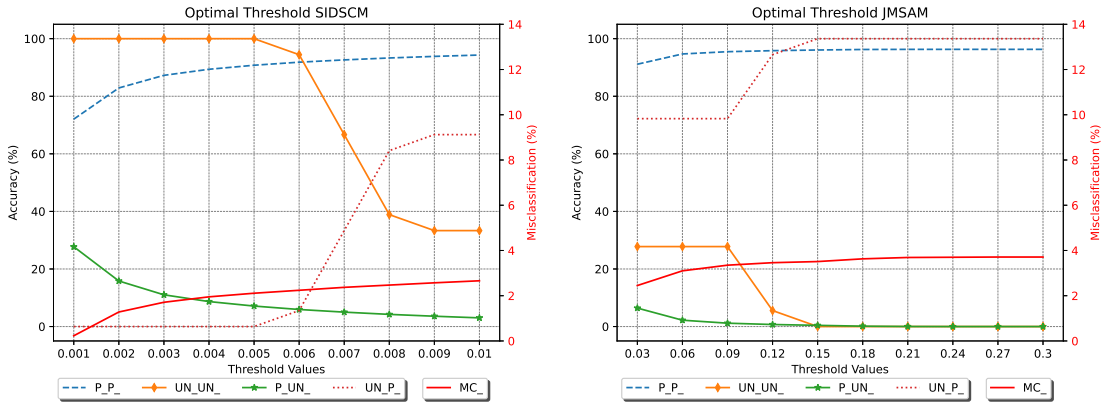
(d) SID



(e) SSS



(f) SID-SAM



(g) SID-SCM (h) (JMSAM)

Figure B.1: Classification accuracy graph of different algorithms at varying threshold values. The graph shows the accuracy of each algorithm in terms of pigment classified as a pigment (P_P_), unknown region classified as unknown (UN_UN_), pigment classified as unknown(P_UN_), unknown classified as a pigment (UN_P_) and pigment classifying as another pigment, i.e., misclassification (MC_).

C Normalized reflectance spectrum of 10 pigments and substrate

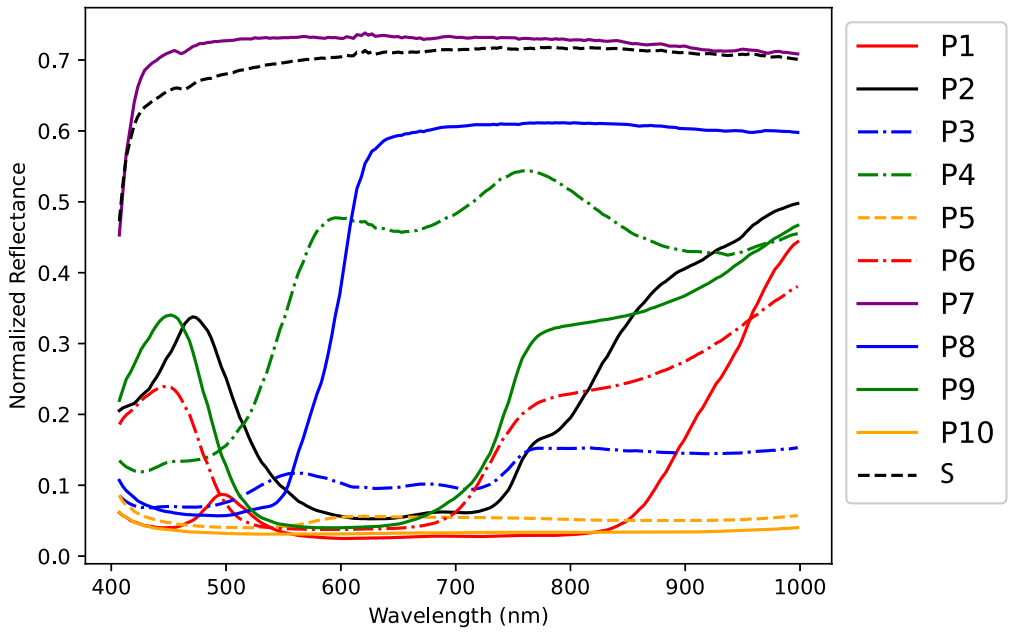
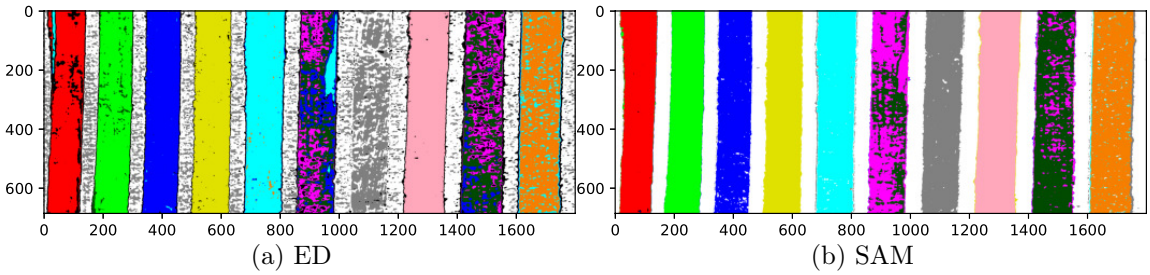
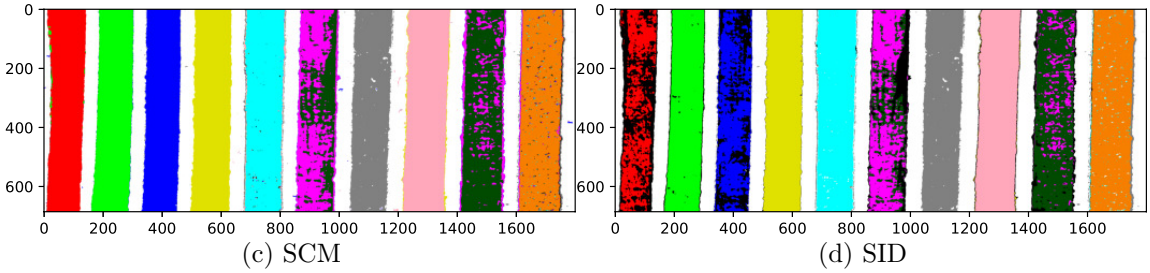


Figure C.1

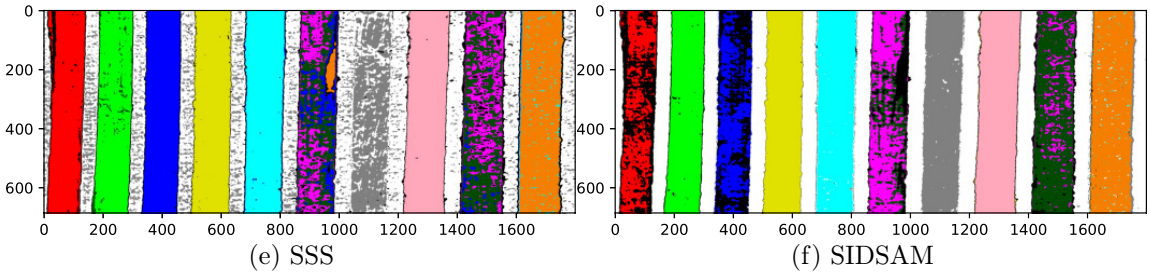
D Classification result for all used algorithms.



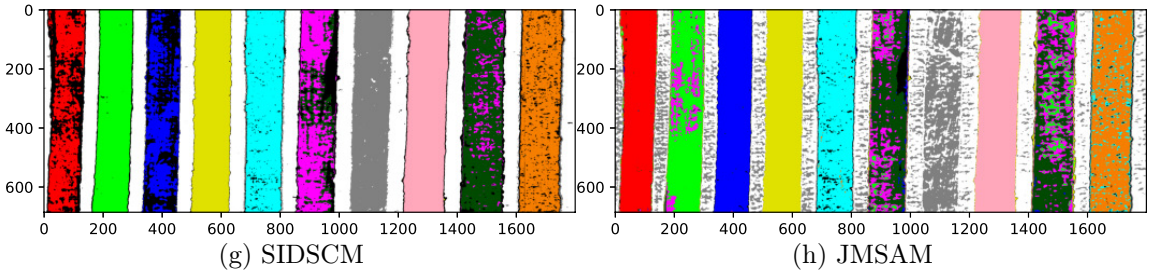
(a) ED (b) SAM



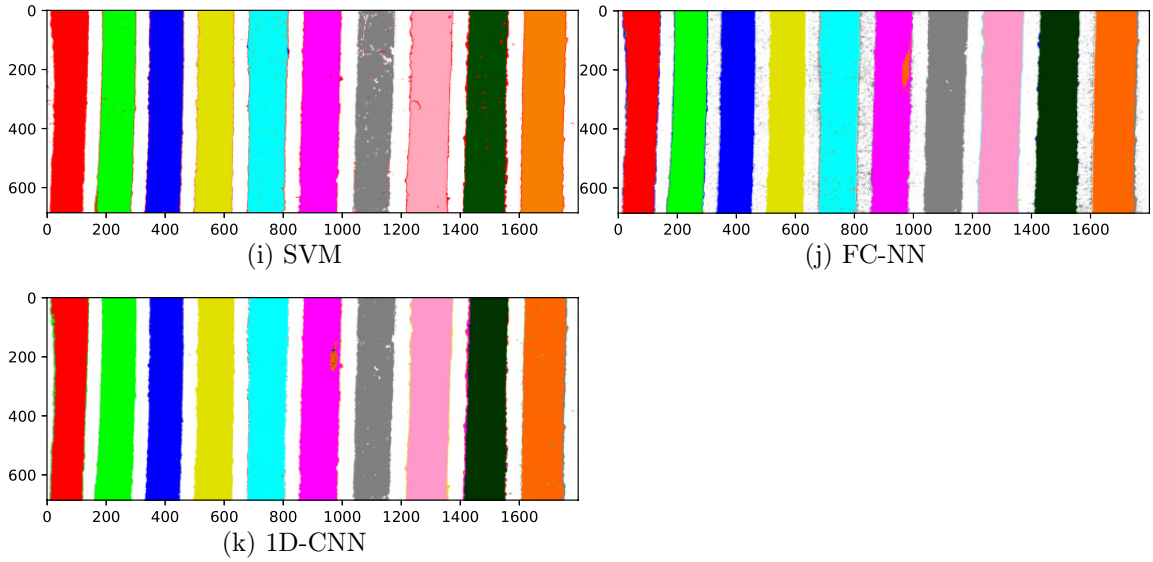
(c) SCM (d) SID



(e) SSS (f) SIDSAM



(g) SIDSCM (h) JMSAM



(i) SVM

(j) FC-NN

(k) 1D-CNN

Figure D.1