Technical University of Crete
Department of Electronic and Computer Engineering

Spectral Prediction From Filtered Color CCD Cameras

Diploma thesis by Dimitris Iliou

Committee: Assoc. Professor Kostas Balas (supervisor)
Professor Minos Garofalakis
Ass. Professor Eutixis Koutroulis

Chania 2011

Acknowledgements

I wish to express my sincere appreciation and thanks to Professor Costas Balas for his guidance and support during the implementation of this diploma thesis.

I am also very grateful to George Epitropou, PhD Candidate and Vasilis Kavadias, MsC Candidate for their guidance and suggestions for the improvement of the current diploma thesis.

Also I would like to thank Professors Minos Garofalakis and Eutixis Koutroulis for their participation in the presentation and the evaluation of this diploma thesis.

Last but not least, I wish to thank my family and my close friends for their advising and support all these years of my studies.

Abstract

Spectral Imaging is a powerful analytical tool and it has been used widely in a long list of different applications, from satellite imaging to biomedical diagnosis, non-destructive analysis etc. The current state of the art mainly includes, scanning spectral imaging system, which require either spatial or spectral scanning of the scene in order to collect the so called Spectral Cube. The Spectral Cube is a stack of tens of hundreds narrow band images. From the Spectral Cube, millions of spectra can be calculated and presented. Due to the time consuming data acquisition and processing procedures associated with the scanning HyperSpectral Imaging Systems, their applications are restricted to static and stationary imaging conditions.

In this study we attempt to reduce that scanning and computational time of HyperSpectral imaging by combining multispectral data acquisition with spectral estimation methods and algorithms. A number of algorithms were evaluated towards this end, and their performance in terms of prediction efficiency and calculation time was assessed. It was found that the Wiener Estimation approach performed quite satisfactory. Particularly, it was found that with less than ten Spectral Bands, Wiener Estimation method predicted the spectra accurately.

These findings are setting the basis for the development of Real-Time HyperSpectral Imagers.

Table of Contents

Chapter Information

Chapter 1 present all the general concepts about spectroscopy, spectral imaging, HyperSpectral imaging, Applications of HyperSpectral Imaging and HyperSpectral Estimation.

Chapter 2 presents all the validation thresholds and the Measures of Spectral Similarities along with the algorithms used.

In ***Chapter 3*** we describe all the experimental methods and all the equipment we used in order to complete the current thesis.

Chapter 4 describes the methodology followed to validate all the algorithms that we have found and all the input test made to find the best performance of the algorithms. Results for each method is provided in order to conclude to the best algorithm.

Chapter 5 summarizes the major findings of this project, and provides a number of recommendations for future work using Real Time Spectral Imaging and expand Spectral Estimation to the IR spectrum.

Chapter 1 : Introduction to HyperSpectral imaging –Estimation

Visual perception of scenes depends on appropriate illumination to visualize objects. The human visual system is limited to a very narrow portion of the spectrum of electromagnetic radiation, called light. In some cases natural sources, such as solar radiation, moonlight, lightning flashes, or bioluminescence, provide sufficient ambient light to navigate our environment. Because humankind was mainly restricted to daylight, one of the first attempts was to invent an artificial light source—fire (not only as a food preparation method).

Computer vision is not dependent upon visual radiation, fire, or glowing objects to illuminate scenes. As soon as imaging detector systems became available other types of radiation were used to probe scenes and objects of interest. Recent developments in imaging sensors cover almost the whole electromagnetic spectrum from x-rays to radiowaves. In standard computer vision applications illumination is frequently taken as given and optimized to illuminate objects evenly with high contrast. Such setups are appropriate for object identification and geometric measurements. Radiation, however, can also be used to visualize quantitatively physical properties of objects by analyzing their interaction with radiation.

Physical quantities such as penetration depth or surface reflectivity are essential to probe the internal structures of objects, scene geometry, and surface-related properties. The properties of physical objects therefore can be encoded not only in the geometrical distribution of emitted radiation but also in the portion of radiation that is emitted, scattered, absorbed or reflected, and finally reaches the imaging system. Most of these processes are sensitive to certain wavelengths and additional information might be hidden in the spectral distribution of radiation. Using different types of radiation allows taking images from different depths or different object properties. As an example, infrared radiation of between 3 and 5 μm is absorbed by human skin to a depth of < 1 mm, while x-rays penetrate an entire body without major attenuation. Therefore, totally different properties of the human body (such as skin temperature as well as skeletal structures) can be revealed for medical diagnosis.

1.1 <u>Spectroscopy/spectrometry</u>

Spectroscopy was originally the study of the interaction between radiation and matter as a function of wavelength (λ). In fact, historically, spectroscopy referred to the use of visible light dispersed according to its wavelength, e.g. by a prism. Later the concept was expanded greatly to comprise any measurement of a quantity as function of either wavelength or frequency. Thus it also can refer to interactions with particle radiation or to a response to an alternating field or varying frequency (ν). A further extension of the scope of the definition added energy (E) as a variable, once the very close relationship E=hν for photons was realized. Spectrometry is the spectroscopic technique used to assess the concentration or amount of a given species. In those cases, the instrument that performs such measurements is a spectrometer or spectrograph.

Spectroscopy/spectrometry is often used in physical and analytical chemistry for the identification of substances through the spectrum emitted from or absorbed by them. Spectroscopy/spectrometry is also heavily used in astronomy and remote sensing. Most large telescopes have spectrometers, which are used either to measure the chemical composition and physical properties of astronomical objects or to measure their velocities from the Doppler shift of their spectral lines.

1.2 Imaging spectroscopy

Imaging spectroscopy (also spectral imaging or chemical imaging) is the application of reflectance spectroscopy to every pixel in a spatial image. Spectroscopy can be used to detect individual absorption features due to specific chemical bonds in a solid, liquid, or gas. Solids can be either crystalline (i.e. minerals) or amorphous (like glasses). Every material is

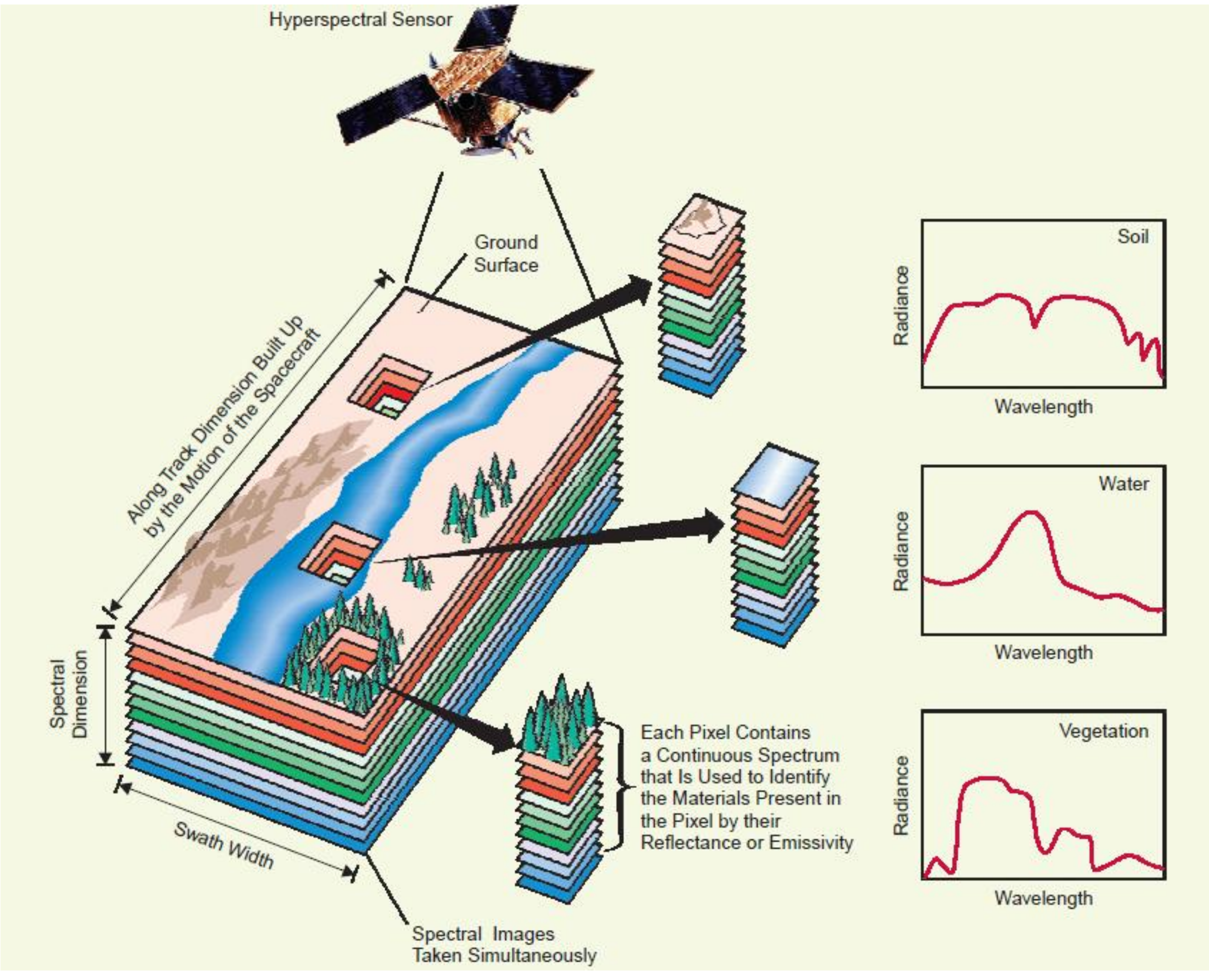


figure 1.1 Principle of (remote sensing) imaging spectroscopy

formed by chemical bonds, and has the potential for detection with spectroscopy. Actual detection is dependent on the spectral coverage, spectral resolution, and signal-to-noise of the spectrometer, the abundance of the material and the strength of absorption features for that material in the wavelength region measured.

In remote sensing situations (figure 1.1), the surface materials mapped must be exposed in the optical surface (e.g., to map surface mineralogy it must not be covered with vegetation), and the diagnostic absorption features must be in regions of the spectrum that are reasonably transparent to the atmosphere (the atmosphere can be corrected for all but the strongest absorptions). The optical surface is the same as what the geologist sees in the field with his or her eyes. Spectroscopy can be used in laboratories on hand samples, in the field with portable field spectrometers (spatial resolution in the millimeter to several meter range), from aircraft, and in the future from satellites. The aircraft systems now operational can image large areas in short time (~2 sq. km per second!), producing spectra for each pixel that can be analyzed for specific absorption bands and thus specific materials. These measurements can then be used for the unambiguous direct and indirect identification of surface materials and atmospheric trace gases, the measurement of their relative concentrations, subsequently the assignment of the proportional contribution of mixed pixel signals (e.g., the spectral unmixing problem), the derivation of their spatial distribution (mapping problem), and finally their study over time (multi-temporal analysis).

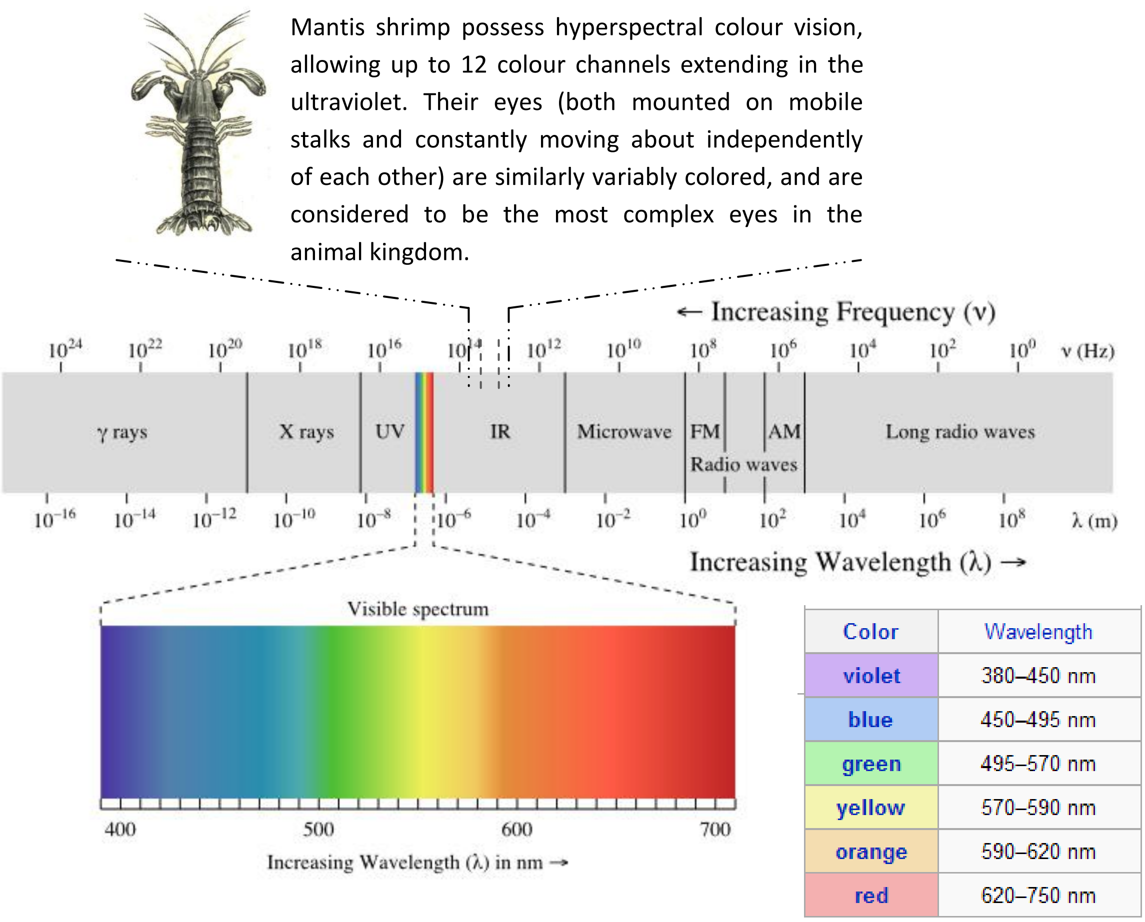
Imaging spectroscopy can be considered as the equivalent of color photography, but each pixel needs to acquire many bands of light intensity data from the spectrum, instead of just the three bands of the RGB color model. More precisely, it is the simultaneous acquisition of spatially coregistered images in many spectrally contiguous bands.

Some spectral images contain only a few image planes of spectral data, while others are better thought of as full spectra at every location in the image. For example, solar physicists use spectroheliograms, images of the Sun built up by scanning the slit of a spectrograph, to study the behavior of surface features on the Sun; such a spectroheliogram may have a spectral resolution of over 100,000 ($\lambda / \Delta\lambda$) and be used to measure local motion (via the Doppler shift) and even the magnetic field at each location in the image plane. The multispectral images collected by the Opportunity rover, in contrast, have only four wavelength bands and hence are only a little more than 3-color images. To be scientifically useful, such measurement should be done using an internationally recognized system of units.

1.3 HyperSpectral Imaging

For the last one hundred years detectors have been developed for radiation of almost any region of the electromagnetic spectrum. Recent developments in detector technology incorporate point sensors into integrated detector arrays, which allows setting up imaging radiometers instead of point measuring devices. Quantitative measurements of the spatial distribution of radiometric properties are now available for (remote) sensing at almost any wavelength.

HyperSpectral imaging collects and processes information from across the electromagnetic spectrum. Unlike the human eye, which just sees visible light, hyperspectral imaging is more like the eyes of the mantis shrimp, which can see visible light as well as from the ultraviolet to infrared. Hyperspectral capabilities enable the mantis shrimp to recognize different types of coral, prey, or predators, all which may appear as the same color to the human eye (figure 1.2).



Color	Wavelength
violet	380–450 nm
blue	450–495 nm
green	495–570 nm
yellow	570–590 nm
orange	590–620 nm
red	620–750 nm

figure 1.2 mantis shrimp's spectacular eye vision spectral range

Humans build sensors and processing systems to provide the same type of capability for application in agriculture, mineralogy, physics, and surveillance and other fields of science. HyperSpectral sensors collect information as a set of 'images'. Each image represents a range of the electromagnetic spectrum and is also known as a spectral band. HyperSpectral sensors look at objects using a vast portion of the electromagnetic spectrum. Certain objects leave unique 'fingerprints' across the electromagnetic spectrum. These 'fingerprints' are known as spectral signatures and enable identification of the materials that make up a scanned object. For example, having the spectral signature for oil helps mineralogists find new oil fields.

The precision of these sensors is typically measured in spectral resolution, which is the width of each band of the spectrum that is captured. If the scanner picks up on a large number of fairly small wavelengths, it is possible to identify objects even if said objects are only captured in a handful of pixels. However, spatial resolution is a factor in addition to spectral resolution. If the pixels are too large, then multiple objects are captured in the same pixel and become difficult to identify. If the pixels are too small, then the energy captured by each sensor-cell is low, and the decreased signal-to-noise ratio reduces the reliability of measured features.

1.4 Single Exposure or Instantaneous Spectral Imagers

In order to record Spectral Images, we use tunable filtered-based Spectral Imaging Systems. Those systems record Spectral Images in a time-sequentially manner and obtain the spectra from post hoc assembly of the time-sequential data. This reveals a major disadvantage of these systems. When we have phenomena that are changing on a time scale that is sorter that the duration required for recording the spectral cube, the SI systems described cannot perform accurately and give accurate Spectral data. Furthermore, the scene recorded from SI systems must be static. Otherwise problems will be created in the co-registration of the Spectral Images, which is needed in order to provide accurate spectra.

There are numerous of applications (ex biomedical or others) that require Spectral Imaging and analysis of transient moving scenes. This is why “Single shot” or “Instantaneous” spectral imagers have been created *[1]*. SE systems have many advantages over SI systems such as : fast acquisition of accurately registered images, high device robustness and reliability, low cost etc. But, there are many trade-offs in order to achieve all that. Due to current technological limitations there is a trade-off between spatial and spectral resolution. That means that SE imagers allow us to capture a small number of spectral images. So, when we have stationary and invariant scenes we use the Spectral Imaging systems described before since the spatial and spectral resolutions are superior to the ones on SE.

However when we want the acquisition of a small number of predetermined bands, SE systems are preferred (for all the advantages described above).

1.5 <u>Color vs. Spectral Imaging</u>

Color Cameras emulate the human vision for color reproduction and are real-time devices since the record those three spectral bands simultaneously. The color imaging device that emulate human vision have three parameters. The Red Green Blue (RGB) parameters. These parameters can be interpreted easily since they model familiar color perception processes. But, unfortunately the human vision has some limitations that are shared by these devices. They allocate the incoming light into the three RGB coordinates, thus missing the important spectral information. So, objects with the same RGB values might have different spectral components. These phenomenon is known as metamerism. So, due to this effect the RGB imaging systems don't have the ability to distinguish material with the same color appearance but different chemical composition. Also, images taken from RGB system have only 3 coordinates for each pixel (R,G, B). In the SI systems for each pixel we have a series of images corresponding to many wavelengths. With these images a spectral cube can be created, which contains the Spectral Information of the object. The nature of the Spectral Cubes' data is (x, y, λ) (spatial coordinates and wavelength). So, as we can see spectral images contain important spectral information of on object – scene, thus the RGB imaging systems lose significant information.

1.6 <u>HyperSpectral Imaging Applications</u>

As we mentioned before, there are numerous of application that HyperSpectral Imaging can be used due to the importance of Spectral Information. HyperSpectral Imaging is nowadays commonly used in biomedicine. For example a study has been made from Georgios N. Stamatas, et al *[2]* in analysis of skin using HyperSpectral Imaging devices in order to produce a reliable quantitative distribution map of chromophores contributing to the color appearance of the skin. A similar work has been made from Daisuke Nakao, et al *[3]*, in order to perform a Real-Time mapping pigmentation in human skin. This is expected to give very useful information about the reproduction of various skin colors as long as various human conditions.

As we mentioned HyperSpectral Imaging can perform a non-destructive analysis. This extremely important, since there are objects that need to be tested without having the possibility to take a physical sample. For example Konstantinos Rapantzikos, et al *[4]* have created a non-destructive analysis method in order to acquire images from manuscripts and detect Palimpsests (twice written manuscripts) which can reveal hidden text under the visible one.

Also, with the use of HyperSpectral Imaging we can perform material or color-pigments classification. In the work if Abdelhameed Ibrahim, et al *[5]* , they use Spectral Information from the Spectral Imager in order to classify the materials of printed circuit boards effectively. In the field of classification we also have the work of Giorgos Epitropou *[6]* where HyperSpectral Imaging is used a long with spectral classification algorithms in order to perform a non-destructive analysis of El Greco's paintings.

From all the above examples, we can see the importance of HyperSpectral Imaging, which provides Spectral Information for the object – scene wanted. Unfortunately, Hyper-Spectral Imaging devices unfortunately are still very heavy and expensive and are restricted to static applications. In addition, these devices cannot be used in real time since they have big acquisition time and post-acquisition processing in order to provide the spectral information of an object/scene. Moreover, data from spectral imaging systems require huge storage.

1.7 <u>HyperSpectral Estimation</u>

As we mentioned before the RGB Imaging devices miss important spectral information. In addition objects with different spectral information may have same RGB values which means that RGB devices cannot detect the difference. So, in the process of digital reproduction of paintings, in order to overcome the problem of metamerism, HyperSpectral Estimation is performed. HyperSpectral Estimation is a method, where the spectrum of an object can by estimated from the RGB equivalent of the object. Many tries have been made in order to save and reconstruct art paintings etc, in digital form. If we only use RGB Imaging systems we will lose important spectral information. So, an effort has been made in order to store the Spectral characteristics of the art-painting in order to perform a much better reproduction of color and from that have better quality of color, since Spectral Reflectance is the most accurate representation of the color of an object.

At first we study the research of Hideaki Haneishi, et al *[7]*, where they try to create a multichannel acquisition system, and with the use of Wiener Estimation method make an accurate approach of the spectral estimation of art paintings. Another research has been made using the same algorithm from P. Stigell et al *[8]*. Using Wiener Estimation method a transformation matrix is calculated using the a priori knowledge, in this case the MacBeth Color Checker (RGB and Spectra Values), and then multiply it with the RGB input of the painting – images they used. The main difference from the other method is that they take the

RGB values of the image as input and they use MacBeth color checker to get a priori knowledge of the spectra under certain lighting. In order to evaluate the quality of the Estimation they used GFC and in order to calculate the errors PSNR and RMSE.

Instead of using the RGB channels as input, Francisco H. Imai, et al *[9]* tried the use of narrow band filters in order to capture and reconstruct Spectral Reflectances. They made a comparison between broadband and narrow band in order to see which of those gives the best results. They found that the two approaches have similar results in the terms of spectral errors. Also, according to their research their results were depended from the target selected. An important remark made is that it is unusual to encounter spectral features which are abrupt or very narrow and that this is encouraging for wide-band approaches.

Another effort was made from Alamin Mansouri et al *[10]* in order to make an accurate Spectral Estimation. In this case we have projection algorithms such as PCA and Wavelets. In their research they use PCA, Wavelets and FFT. On this study the main goal is to derive the Basis functions and then use the formentioned algorithms. In this case, they also use MacBeth Color Checker as a priori knowledge to train the basis and then they use the RGB input of the images in order to make the Spectral Estimation. In addition they tried Multispectral Images as input in order to evaluate the spectrum of the image. Wavelets gave the best results in this research but in order to get these results post – estimation calculation was needed in order to smooth the data of the wavelets. The evaluation of the results were done with the use of GFC.

An improvement done for this work was from Alamin Mansouri et al *[11]* where they tried to create an adaptive PCA algorithm in order to minimize the difference between the estimated and the original spectra. They again derive a basis function as in their previous study and then the estimate the spectra. Then, using the GFC they change the training by removing the spectra that don’t give good results, and then recalculate the basis function etc.

Another use of PCA was performed from Francisco H. Imai et al *[12]* where, they the technique of spectral reflectance estimation is based on a priori spectral analysis and principal component analysis. The eigenvectors produced from PCA are coupled with the trichromatic digital cameras combined with absorption filters for the Estimation process. Through this system, they managed to create an efficient spectral reconstruction system.

From the previous work we saw that linear Projection algorithms can be used. So we researched for some of these algorithms in order to test them in the same method and see if we can get the same results. We came up with Hilbert Transformation *[13]* and since we came across PCA we also found Singular Value Decomposition *[14]*. Finally we also found the Discrete Cosine Transform (DCT) *[15]*.

In all the above studies Goodness Of Fit (GFC)*[16-17]* was used in order to validate the results from the estimated spectra to the original spectra. So, in our study we also used the GFC in order to evaluate our results. Also, another important remark is that all these studies evaluate the estimation for the visible spectrum. Most objects have smooth spectral reflectance on the visible spectrum. The existence of steep peeks and deeps can make the estimation process very difficult. That's why we study Spectral Estimation on the visible Spectrum in this diploma thesis. Of course, there are many more validation metrics that we can use developed at Munsell Color Science Laboratory by Francisco H. Imai et al *[17]*.

In addition is very important to know that the choice of training set used in order to perform Spectral Estimation is of crucial importance. In recent studies, Shimano and Hironaga *[18]* verified the choice of the training samples is very important to accurate reflectance estimation. Another remark that can be made in this point is from the work of Eva M. Valero et al *[19]*,where the difference between rural and urban is pointed out. According to this work different training sets are needed in each case in order to perform accurate Spectral Estimation, since natural environments have many complex variations in spatial structure as long as uncontrolled illumination, which can cause many problems during the acquisition and estimation process.

So we understand that the lighting conditions are also of great importance. In all the previous research done it is pointed out that all the object (Input and a priori data) must be taken under the same lighting. That's a strict principle that we followed through our experimental procedures.

Furthermore in the study of Hui-Kiang Shen et al *[20]* we saw that the use of different devices for spectral estimation can cause cross-media instrument metamerism due to different optical and mechanical design. The former proposes a method in order to correct this phenomenon in reflectance estimation from multispectral imaging.

Chapter 2 : Algorithms and Validation Thresholds

2.1 Linear Algebra

In most cases we use linear algebra to represent and solve equations for Spectral Estimation. For that reason we define some linear algebra rules.

Given a matrix A:

- Transpose of A : $\left[A^T\right]_{i,j} = \left[A\right]_{j,i}$
- The product of an m×p matrix A with a n p×n matrix B is an m×n matrix denoted AB whose entries are

$$(AB)_{i,j} = \sum_{k=1}^{p} A_{i,k} B_{k,j}$$

Where 1 ≤ i ≤ m is the row index and 1 ≤ j ≤ n is the column index.

- Inversion of A : $A^{-1}A = I$ where for example a 2x2 matrix inversion is

$$A^{-1} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} = \frac{1}{ad - bc}\begin{pmatrix} d & -b \\ -c & a \end{pmatrix}$$

- Pseudo-inverse matrix D is : $D^{+} = \left(D^T D\right)^{-1} D^T$

2.2 Measures of Spectral Similarities

Peak Signal to noise Ratio (PSNR)

The phrase peak signal-to-noise ratio, often abbreviated PSNR, is an engineering term for the ratio between the maximum possible power of a signal and the power of corrupting noise that affects the fidelity of its representation. Because many signals have a very wide dynamic range, PSNR is usually expressed in terms of the logarithmic decibel scale.

$$PSNR = 10\log\left(\frac{2^{bpp} - 1}{RMSE}\right)$$

Root Mean Square Error (RMSE)

In statistics, the mean square error or MSE of an estimator is one of many ways to quantify the difference between an estimator and the true value of the quantity being estimated. MSE is a risk function, corresponding to the expected value of the squared error loss or quadratic loss. MSE measures the average of the square of the "error." The error is the amount by which the estimator differs from the quantity to be estimated. The difference occurs because of randomness or because the estimator doesn't account for information that could produce a more accurate estimate.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} \Delta x_i^2}{n}}$$

Where in our study n is the number of spectral channels and x the spectral reflectance vector of one pixel. Δx_i^2 is the squared difference of the i-th channel values of 2 spectra.

Goodness of fit (GFC)

GFC is defined as the cosine of the angle between the recovered signal S^ and original signal S, thus

$$GFC = \frac{\sum_{\lambda=400}^{700} \hat{S}(\lambda) S(\lambda)}{\left(\sum_{\lambda=400}^{700} \hat{S}(\lambda)^2\right)^{1/2} \left(\sum_{\lambda=400}^{700} S(\lambda)^2\right)^{1/2}}$$

Spectral Angle Mapper (SAM)

SAM is a non parametric supervised classifier. The SAM algorithm considers every pixel of the spectral image as a vector, whose length corresponds to the brightness that this pixel has, and the direction of the vector features the spectral characteristics of the pixel. It can be calculated from:

$$\langle x, y \rangle = \|x\| \|y\| \cos(x, y) \rightarrow$$

$$\theta(x, y) = SAM(x, y) = \cos^{-1}\left(\frac{\langle x,y \rangle}{\|x\| \|y\|}\right),\ 0 \leq \theta \leq \frac{\pi}{2}$$

SAM is used for the calculation of the angle between the pixel in question and the reference vectors, uses only the direction of the vector and not its length. That’s why SAM is independent of lighting. Also, it is independent to the multiplications of a vector with a natural number since it only increases its length and doesn’t change the angle. So, SAM is a non-prosthetic distance function.

Euclidean Distance

In mathematics, the Euclidean distance or Euclidean metric is the "ordinary" distance between two points that one would measure with a ruler, and is given by the Pythagorean formula. By using this formula as distance, Euclidean space (or even any inner product space) becomes a metric space. The associated norm is called the Euclidean norm. Older literature refers to the metric as Pythagorean metric.

The Euclidean distance between points p and q is the length of the line segment connecting them ($\overline{\mathbf{pq}}$).

In Cartesian coordinates, if p = (p1, p2,..., pn) and q = (q1, q2,..., qn) are two points in Euclidean n-space, then the distance from p to q, or from q to p is given by:

$$\mathrm{d}(\mathbf{p},\mathbf{q}) = \mathrm{d}(\mathbf{q},\mathbf{p}) = \sqrt{(q_1-p_1)^2 + (q_2-p_2)^2 + \cdots + (q_n-p_n)^2} = \sqrt{\sum_{i=1}^{n}(q_i-p_i)^2}.$$

2.3 Gretag Macbeth® Color Checker®

The Macbeth ColorChecker® is a unique test pattern scientifically designed to help determine the true color balance or optical density of any color rendition system. It is an industry standard that provides a non-subjective comparison with a “test pattern” of 24 scientifically prepared colored squares. Each color square represents a natural object—human skin, foliage, blue sky, etc, providing a qualitative reference to quantifiable values. Each color will reflect light in the same way in all parts of the visible spectrum, thus maintaining color consistency over different illumination options. Some applications include spectroscopy, machine vision, photography, graphic arts, electronic publishing, and television. In our study we use Macbeth ColorChecker mostly as a priori samples. From now on when we present plots we know that Pad 1 is the upper left pad and we increase the number by row. So the last pad is the Black on bottom right which is Pad 24.

2.4 Algorithms

As we will see there are many methods for spectral-recovery using RGB digital cameras. In our study we have:

- Wiener Estimation Method

- PCA
- Wavelets
- SVD
- Fast Fourier Transform
- DCT

Wiener Estimation Method

The first algorithm we examined was Wiener Estimation Method in Estimating of Spectral Reflectance from RGB Images *[8]*. Wiener Estimation Method is based on the use of a priori knowledge. This comes from a Gretag Macbeth standard color checker with 24 color pads. Our goal here is to make estimations from low-dimensional data to high dimensional data as reflectance spectra.

First we need to calculate the response of a digital camera. The response $v_i(x,y)$ in spatial coordinates (x,y) with i-th color filter can be calculated from the following equation:

$$v_i(x,y) = \int t_i(\lambda) E(\lambda) S(\lambda) r(x,y:\lambda)\, d\lambda, \qquad i = 1,..3 \quad (1)$$

Where $t_i(\lambda)$ is the permeability of the i-th filter, $E(\lambda)$ is the spectrum of the illuminant, $S(\lambda)$ is the sensitivity of the camera and $r(x,y:\lambda)$ is the reflectance spectrum in spatial coordinates (x,y). Also i goes to three since we used a RGB camera.

As we mentioned before is useful to represent equations with linear algebra. That helps us also because in SI, data are collected in a spectral cube which is a set of pixel vectors. So we write the above equation as

$$v = Fr \quad (2)$$

Where v is an m element column vector and r is a k element column vector which corresponds to the reflectance spectrum of one pixel of an image. F is a linear (mxk) matrix which relates the two vectors mentioned above.

To solve the equation (2) we use the linear Wiener estimation method $r_{est} = Gv$ where G is the estimation matrix. G is represented as : $G = R_{rv} R_{vv}^{-1}$ where R_{rv}, R_{vv} are correlation matrices.

The spectral image data $f\left(x,y,\lambda\right)$ are calculated from the equation: $f\left(x,y,\lambda\right)=Gv$ where G, v are mentioned above. With vector v we can include higher order pixel values. That means that instead of using only RGB values, we also use $\left[R*G,R*B,G*B,R^2,G^2,B^2\right]$ and a constant value 1. So we can use more terms in the spectral estimation.

For Measures for recovery similarity and error measure we use the methods introduces in section 1.2. Also in this method, in testing noise of the camera was taken into account.

Principal Component Analysis

Principal component analysis (PCA)*[21]* is the best, in the mean-square error sense, linear dimension reduction technique. Being based on the covariance matrix of the variables, it is a second-order method. In various fields, it is also known as the singular value decomposition (SVD), the Karhunen-Loeve transform, the Hotelling transform, and the empirical orthogonal function (EOF) method. In essence, PCA seeks to reduce the dimension of the data by finding few orthogonal linear combinations (the PCs) of the original variables with the largest variance. In other words, the goal of principal component analysis is to compute the most meaningful (orthonormal) basis to re-express a noisy data set. The hope is that this new basis will filter out the noise and reveal hidden structure.

By assuming linearity PCA seeks a linear combination of the original basis that best re-express a given dataset. The first component, s_1, is the linear combination with the largest variance. We have $s_1 = \boldsymbol{x}^T\boldsymbol{w_1}$, where the p-dimensional coefficient vector $\boldsymbol{w}$=[w_1 w_2 ... w_p] *solves*

$$w_1 = argmax_{||w=1||} Var\{\boldsymbol{x}^T\boldsymbol{w}\}$$

The second PC is the linear combination with the second largest variance and orthogonal to the first PC, and so on. There are as many PCs as the number of the original variables. For many datasets, the first several PCs explain most of the variance, so that the rest can be disregarded with minimal loss of information. Since the variance depends on the scale of the variables, it is customary to first standardize each variable to have mean zero and standard deviation one. After the standardization, the original variables with possibly different units of measurement are all in comparable units. Assuming a standardized data with the empirical covariance matrix

$$\Sigma_{pxp} = \frac{1}{n-1}\boldsymbol{XX}^{T}.$$

Note that XX^T is a symmetric matrix so it can be diagonalized by its orthonormal eigenvectors. Thus

$XX^T = U\Lambda U^T$,

Where $\Lambda = diag[\lambda_1\ \lambda_2\ \ldots\ \lambda_p]$ is the diagonal matrix of the ordered eigenvalues $\lambda_1 < \lambda_2 < \ldots < \lambda_p$ and U is a p x p orthogonal matrix containing the eigenvectors. It can be shown [22] that the PCs are given by the p rows of the p x n matrix S, where $S = U^T X$ (the weight matrix W is given by U^T). The subspace spanned by the first k eigenvectors has the smallest mean square deviation from X among all subspaces of dimension k. Another property of the eigenvalue decomposition is that the total variation is equal to the sum of the eigenvalues of the covariance matrix,

$$\sum_{i=1}^{p} \mathrm{Var}(\mathrm{PC_i}) = \sum_{i=1}^{p} \lambda_\iota = \sum_{i=1}^{p} \mathrm{trace}(\Sigma)$$

and that the fraction $\sum_{i=1}^{\kappa} \lambda_\iota/\mathrm{trace}(\Sigma)$ gives the cumulative proportion of the variance explained by the first κ PCs.

Performing PCA is quite simple in practice:

1 .Organize a data set as an mxn matrix, where m is the number of measurement types and n is the number of trials.

2. Subtract of the mean for each measurement type or row xi .

3. Calculate the eigenvectors (or the SVD) of the covariance.

One benefit of *PCA i*s that we can examine the variances ($\mathrm{Var}(\mathrm{PC_i})$) associated with the principle components. Often one finds that large variances associated with the first k<p principal components, and then a precipitous drop-off. One can conclude that most interesting dynamics occur only in the first k dimensions. Both the strength and weakness of *PCA* is that it is a *non-parametric* analysis.

The interpretation of the PCs can be difficult at times. Although they are uncorrelated variables constructed as linear combinations of the original variables, and have some desirable properties, they do not necessarily correspond to meaningful physical quantities. In some cases, such loss of interpretability is not satisfactory to the domain scientists.

Wavelets

In mathematics, the Haar wavelet is a certain sequence of rescaled "square-shaped" functions which together form a wavelet family or basis. Wavelet analysis is similar to Fourier analysis in that it allows a target function over an interval to be represented in terms of an orthonormal function basis. The Haar sequence is now recognised as the first known wavelet basis and extensively used as a teaching example in the theory of wavelets.

The Haar sequence was proposed in 1909 by Alfréd Haar. Haar used these functions to give an example of a countable orthonormal system for the space of square-integrable functions on the real line. The study of wavelets, and even the term "wavelet", did not come until much later. As a special case of the Daubechies wavelet, it is also known as D2.

The Haar wavelet is also the simplest possible wavelet. The technical disadvantage of the Haar wavelet is that it is not continuous, and therefore not differentiable. This property can, however, be an advantage for the analysis of signals with sudden transitions, such as monitoring of tool failure in machines.

The Haar wavelet's mother wavelet function ψ(t) can be described as

$$\psi(t) = \begin{cases} 1 & 0 \leq t < 1/2, \\ -1 & 1/2 \leq t < 1, \\ 0 & \text{otherwise.} \end{cases}$$

Its scaling function ϕ(t) can be described as

$$\phi(t) = \begin{cases} 1 & 0 \leq t < 1, \\ 0 & \text{otherwise.} \end{cases}$$

Singular Value Decomposition

In linear algebra, the singular value decomposition (SVD) is a factorization of a real or complex matrix, with many useful applications in signal processing and statistics.

Formally, the singular value decomposition of an m×n real or complex matrix M is a factorization of the form

$$M = U\Sigma V^*,$$

where U is an m×m real or complex unitary matrix, Σ is an m×n diagonal matrix with nonnegative real numbers on the diagonal, and V* (the conjugate transpose of V) is an n×n real or complex unitary matrix. The diagonal entries Σi,i of Σ are known as the singular values of M. The m columns of U and the n columns of V are called the left singular vectors and right singular vectors of M, respectively.

The singular value decomposition and the eigen decomposition are closely related. Namely:

- The left singular vectors of M are eigenvectors of MM * .
- The right singular vectors of M are eigenvectors of M * M.
- The non-zero singular values of Σ are the square roots of the non-zero eigenvalues of M * M or MM * .

Applications which employ the SVD include computing the pseudoinverse, least squares fitting of data, matrix approximation, and determining the rank, range and null space of a matrix.

Fast Fourier Transform

A fast Fourier transform (FFT)*[22]* is an efficient algorithm to compute the discrete Fourier transform (DFT) and its inverse. There are many distinct FFT algorithms involving a wide range of mathematics, from simple complex-number arithmetic to group theory and number theory;

A DFT decomposes a sequence of values into components of different frequencies. This operation is useful in many fields but computing it directly from the definition is often too slow to be practical. An FFT is a way to compute the same result more quickly: computing a DFT of N points in the naive way, using the definition, takes O(N2) arithmetical operations, while an FFT can compute the same result in only O(N log N) operations. The difference in speed can be substantial, especially for long data sets where N may be in the thousands or millions—in practice, the computation time can be reduced by several orders of magnitude in such cases, and the improvement is roughly proportional to N / log(N). This huge improvement made many DFT-based algorithms practical; FFTs are of great importance to a wide variety of applications, from digital signal processing and solving partial differential equations to algorithms for quick multiplication of large integers.

The most well known FFT algorithms depend upon the factorization of N, but (contrary to popular misconception) there are FFTs with O(N log N) complexity for all N, even for prime N. Many FFT algorithms only depend on the fact that $e^{-\frac{2\pi i}{N}}$ is an Nth primitive root of unity, and thus can be applied to analogous transforms over any finite field, such as number-theoretic transforms.

Since the inverse DFT is the same as the DFT, but with the opposite sign in the exponent and a 1/N factor, any FFT algorithm can easily be adapted for it.

Discrete Cosine Transform

A discrete cosine transform (DCT) expresses a sequence of finitely many data points in terms of a sum of cosine functions oscillating at different frequencies. DCTs are important to numerous applications in science and engineering, from lossy compression of audio (e.g. MP3) and images (e.g.JPEG) (where small high-frequency components can be discarded), to spectral methods for the numerical solution of partial differential equations. The use of cosine rather than sine functions is critical in these applications: for compression, it turns out that cosine functions are much more efficient (as described below, fewer are needed to approximate a typical signal), whereas for differential equations the cosines express a particular choice of boundary conditions.

In particular, a DCT is a Fourier-related transform similar to the discrete Fourier transform (DFT), but using only real numbers. DCTs are equivalent to DFTs of roughly twice the length, operating on real data with even symmetry (since the Fourier transform of a real and even function is real and even), where in some variants the input and/or output data are shifted by half a sample. There are eight standard DCT variants, of which four are common.

The most common variant of discrete cosine transform is the type-II DCT, which is often called simply "the DCT"; its inverse, the type-III DCT, is correspondingly often called simply "the inverse DCT" or "the IDCT". Two related transforms are the discrete sine transform (DST), which is equivalent to a DFT of real and odd functions, and the modified discrete cosine transform (MDCT), which is based on a DCT of overlapping data.

2.5 Validation Thresholds

As we have seen from all the work previously done on spectral estimation, GFC is used in order to evaluate the Estimated result. We created some thresholds of success using 2 criteria. The one will be GFC and the other will be spectral angle. Since, we are interested purely in spectra and not in color reproduction we want the spectral information that we estimated to be as accurate as possible. Thus, the following three thresholds:

1st Threshold:

- GFC >0.9900
- Spectral Angle < 0.2 rads

2nd Threshold:

- GFC >0.9950

- Spectral Angle < 0.1 rads

3rd Threshold :

- GFC >0.9990
- Spectral Angle < 0.1 rads

The GFC comes from the work of Mansouri et al [10] where they made the above categories. So we have that:

- Estimation Threshold < 1st threshold: Satisfactory or poor
- Estimation Threshold ≥ 1st threshold : Good estimation
- Estimation Threshold ≥ 2nd threshold : Very good estimation
- Estimation Threshold ≥ 3rd threshold : Excellent estimation

We also use Spectral Angle in order to get as much information as we can about the difference between the estimated and the original spectrum. That why our thresholds consist of a combination of the two metrics.

Chapter 3 : Experimental Materials and Methods

In the previous sections we mentioned the algorithms that we can use for Hyper Spectral Estimation. The first thing that we had to do was to validate that these algorithms truly work and give the results presented in the papers. After validating the algorithms we had to make a choice, based on specific Thresholds, on which algorithm is best for the Hyper Spectral Estimation.

In all cases the a priori knowledge needed, as well as the input for the estimation, came from an RGB camera and as it concerns the spectra data from a spectrometer. To validate the results the algorithm we use The GretagMacBeth ColorChecker.

3.1 RGB and Spectra Data Acquisition

The first thing that we needed to do is to obtain the data needed. That was

- RGB pictures of MacBeth
- Reflectance of Macbeth

For the measurements needed we used a series of devices.

- RGB Dragonfly Camera
- Spectrometer
- MUSIS HyperSpectral Camera

Dragonfly RGB camera (Image 1) is CCD board level camera specifically designed for industrial machine vision tasks.

Image 1 : Dragonfly Camera

A spectrometer (spectrophotometer, spectrograph or spectroscope) is an instrument used to measure properties of light over a specific portion of the electromagnetic spectrum, typically used in spectroscopic analysis to identify materials. The variable measured is most often the light's intensity but could also, for instance, be the polarization state. The independent variable is usually the wavelength of the light or a unit directly proportional to the photon energy, such as wavenumber or electron volts, which has a reciprocal relationship to wavelength. A spectrometer is used in spectroscopy for producing spectral lines and measuring their wavelengths and intensities. Spectrometer is a term that is applied to instruments that operate over a very wide range of wavelengths, from gamma rays and X-rays into the far infrared. If the instrument is designed to measure the spectrum in absolute units rather than relative units, then it is typically called a spectrophotometer. The majority of spectrophotometers are used in spectral regions near the visible spectrum.

We had to take measurements for all 24 pads of the Gretag MacBeth ColorChecker. It was very important that all measurement were taken under the same lighting conditions so we wouldn't have any variations between measurements. So we set up the RGB camera and the Spectrometer and we begun to take measurements for each pad. Also, we measured the RGB and reflectance spectra using the MUSIS HyperSpectral Camera. The reason we did that was to see if we will get better results by using only one device than two. That's because MUSIS can take both RGB and Spectra Images. In our work we use the Ocean Optics USB 2000+ Spectrometer (Image 2).

Image 2 : Spectrometer

Finally we used the MUSIS HyperSpectral Camera. MUSIS is a HySI system capable of real time spectral imaging (both reflectance and fluorescence) with high spectral resolution and high throughput ratio, D.Anglos et al *[23]*, C.Balas *[24]* developed an all-optical imaging monochromator functioning as an electronically tunable narrow band pass optical filter. Displacement of the optical elements of the latter, results in the tuning of the imaging wavelength, which is performed with the aid of electromechanical manipulators controlled from the PC via microcontroller. Mu.SIS HS' (Image 3) technical features are:

- Spectral imaging acquisition of 5nm full width half maximum (FWHM), performing in 34 spectral bands of about 20nm each, in the spectral range 360nm (Ultraviolet)–1550nm (Near Infrared).
- Real time capturing & displaying images with an analysis of 1600x1200 pixels.
- Minimum transmittance is 40% across its operational spectral range, which determines the high throughput of the developed monochromator.
- Tuning spectral range of the filtering system is matched with the responsivity spectral range of the charge coupled device (CCD) image sensor, with the capability of extending to longer wavelengths, up to the mid-infrared range (photocathode).
- A megapixel CCD camera, for feeding back the monochromator signal, based on the IEEE-1394 data transferring protocol, capable of acquiring images at a rate of 15 frames/s at full resolution and of more than 30 frames/sat VGA resolution.
- A special calibration procedure *[25]* is executed before any imaging procedures, compensating for the wavelength dependence of the response of the electro optical parts of the system, such as CCD, illuminators, etc, thus ensuring the full exploitation of the CCD's dynamic range.

- Operating in imaging mode, an image at each wavelength band is acquired while, in spectroscopy mode, a fully resolved diffuse reflectance and/or fluorescence spectrum per image pixel can be recorded (image spectral cube). The combination of spectral and color imaging with calibration enables the system to operate as either Imaging Spectrometer or Imaging Colorimeter.

Image 3 : MUSIS HS Camera

The main difference we have between the MUSIS HS and the Spectrometer is that MUSIS has an acquisition integral of 2nm in contrast with Spectrometer, which has lower than 1nm. This affects our data in the following way: if we want to use the data from MUSIS we must first make a smoothing process of the data. Also when using Spectrometer we need to fit the data so we remove all kinds of noise the Spectrometer might have. From all the spectral data we need the values from 420nm to 700nm (Visible Spectrum). Our work could be expanded to the near infrared (work in progress).

Finally, one of the main reasons we wanted to use the MUSIS HS camera was that we would have all the devices needed in one camera. So we didn't have to make different calibration for each device or have a fault in aligning the second device in the same spot as the first. So, to keep it sort, MUSIS HS makes our work a lot easier.

Chapter 4: Validation of the Algorithms And Experimental Results

4.1 Validation of the algorithms

After obtaining the a priori data needed to use in the algorithms we proceedws to the validation process. The validation process is done with the use of MacBeth ColorChecker.

After estimating each pad we compared the result with the spectra measured with MUSIS HS or Spectrometer.

For the estimation process we didn't use the whole picture that is captured. We only used the center pixel of each pad. That because we needed to verify that the algorithms works before we tried to produce the spectral cube of the image captured. So all the values we have for the RGB and Triple Band are from the center pixel of each pad.

For starters, we processed and constructed the a priori and input data needed. For the RGB values we constructed our data with the two following ways. We took the central RGB value of each pad and thus creating a matrix of 24x3 elements. Another way was to create the same matrix but not by obtaining only the centre RGB value but by choosing a part of the image (preferably center) and calculating the average of all the RGB values in that part.

Then, we created the matrix with the spectral values. Since we smoothed the MUSIS HS data and we have a 1nm integral with the spectrometer, we can choose the integral we want to have in our a priori spectral data. We most commonly use in our project a 10 nm integral, from 420nn-700nm, creating a 29x24 matrix. Smaller integral gives more a priori data for the algorithms. Furthermore, it is useful to remark that data from Spectrometer and MUSIS can vary a bit (Images 4), but with great significance to the results.

After studying the algorithms we found that we have 3 different overall categories:

- Basis Projection Algorithms (With Spectral Sensitivity of the camera)
- Basis Projection Algorithms (Without Spectral Sensitivity of the camera)
- Wiener Estimation Method

For the validation process we needed to test all the algorithms and select the best one for our work. The Thresholds for that purpose were the results (compared with plots and estimation metrics) and time.

Since we need to have the spectral sensitivity of our cameras for the algorithms, we measured all the Spectral Sensitivities needed. So, we quote in the appendix the Spectral Sensitivities of the Cameras we used (MUSIS HS, DragonFly).

The first think we wanted to clear was the usefulness of the spectral sensitivity. The first problem encountered with those algorithms was that in many cases the data were needed in different forms (so no data universality between methods). Also after checking both types of algorithms we saw that there aren't that many differences. So, after the first round of testing we eliminated the Basis Projection Algorithms with Spectral Sensitivity since we got the same results but with the use of more data.

The first algorithm we verified was the Wiener Estimation Method. When using 3 Terms estimation (R, G and B in our case) according to Wiener we can use polynomials to make the algorithm more accurate. So we had, as described above, 3, 7 and ten terms. We wanted also to check, which number of terms we should use when using Wiener. Thus, in the results presented we can see the estimation for every case of terms (Image 5).

Image 4 : MUSIS HS Vs Spectrometer

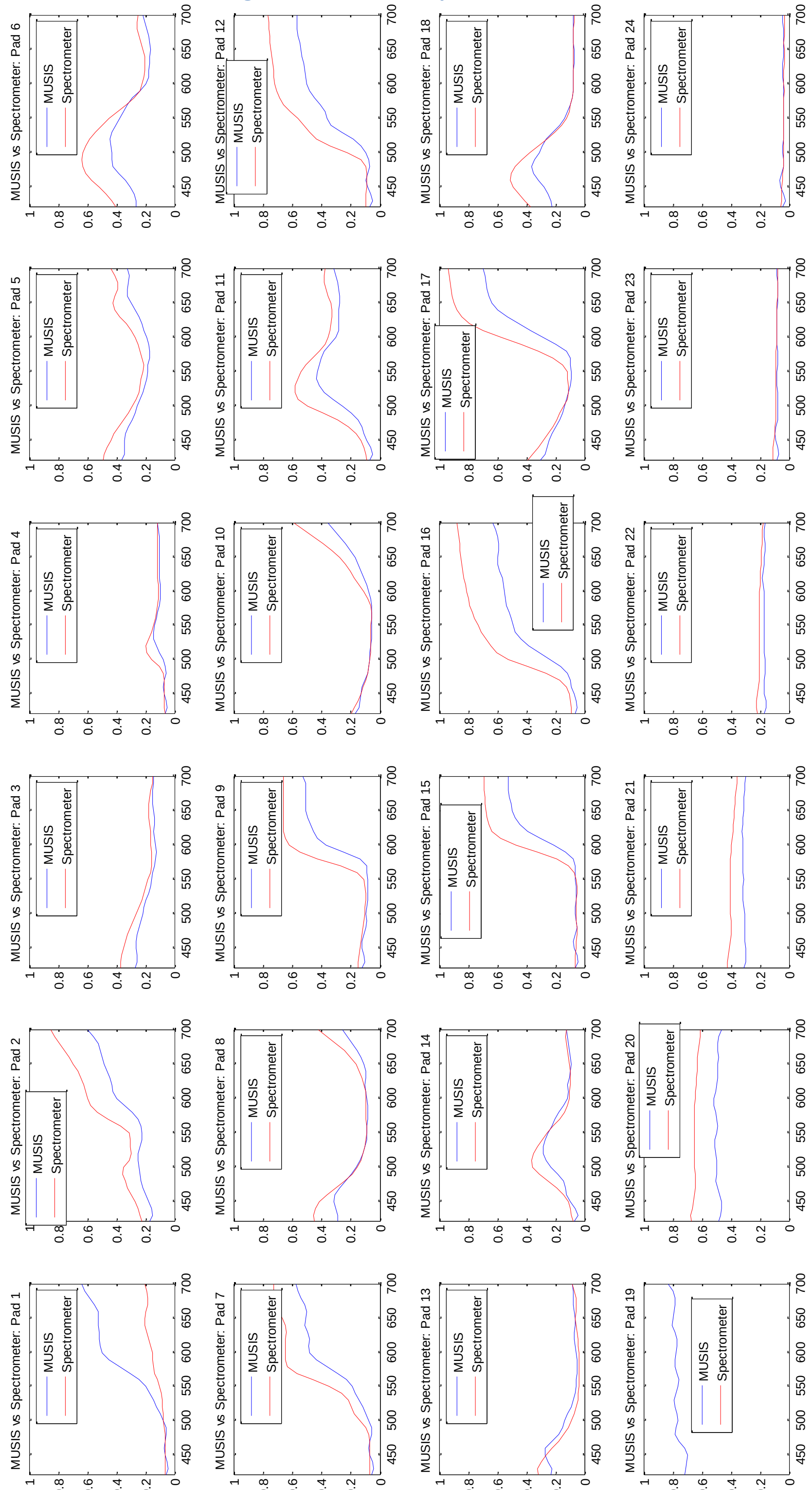


Image 5 : Wiener Estimation Method 3 Terms

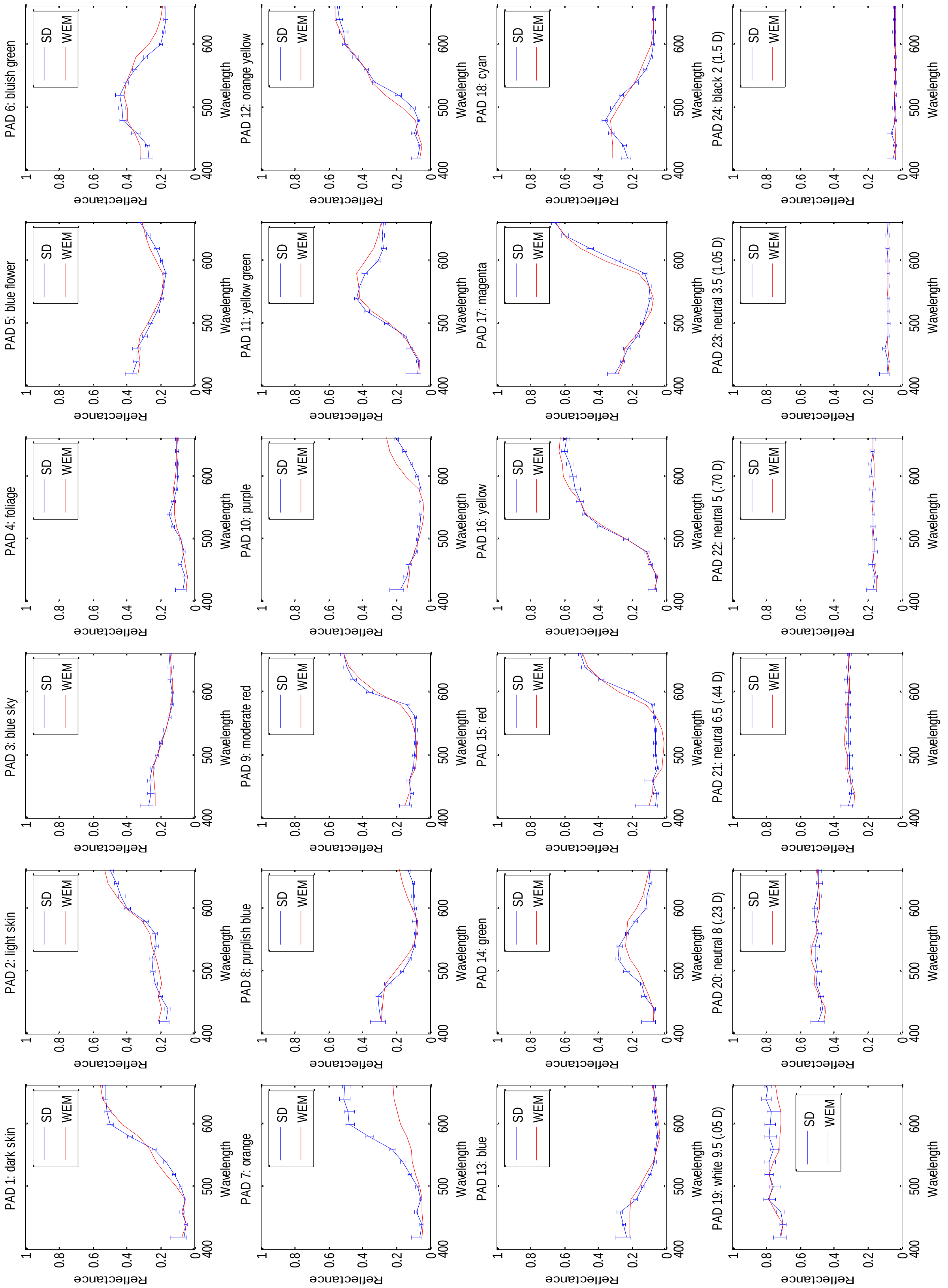


Image 6 : Wiener Estimation Method 7 Terms

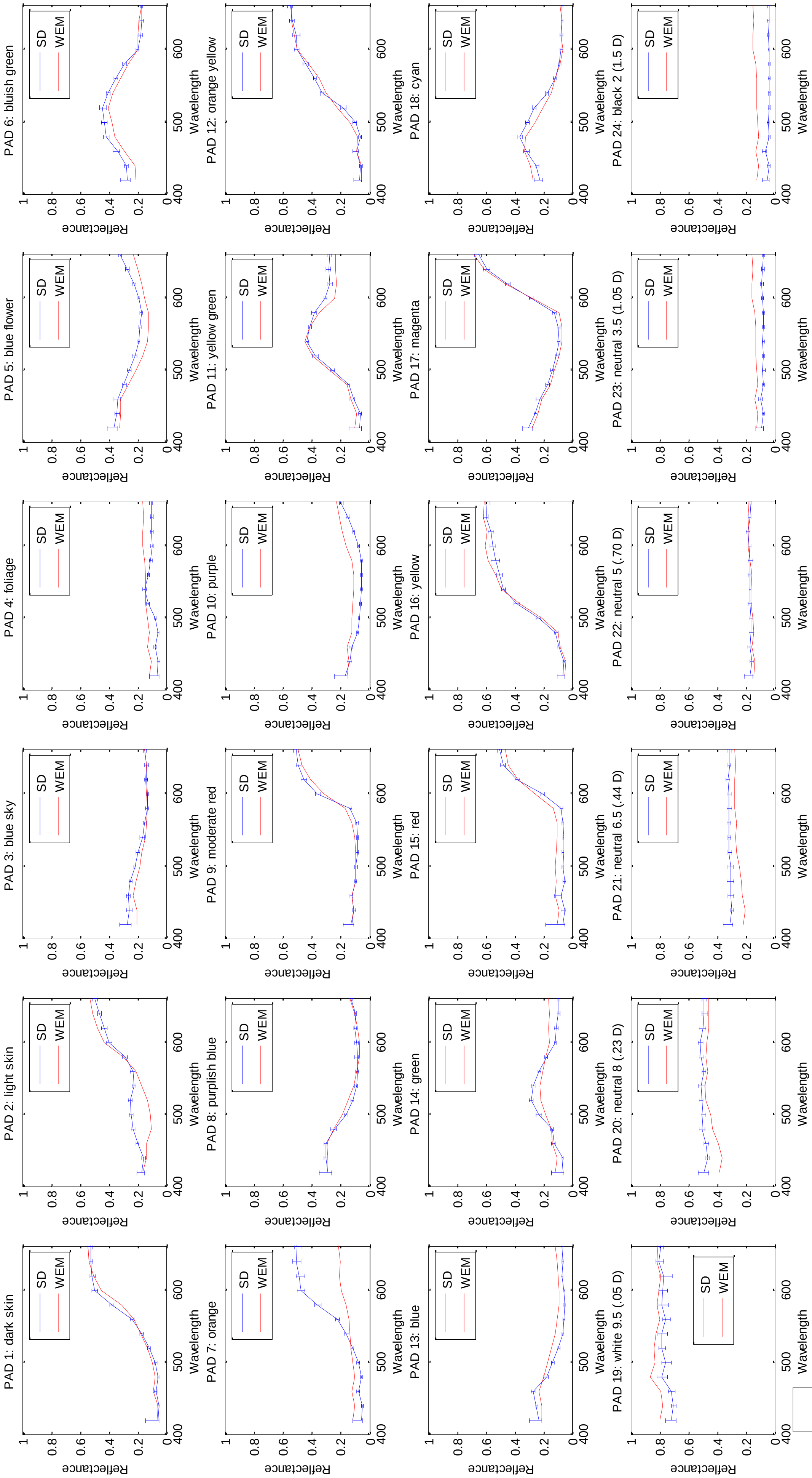
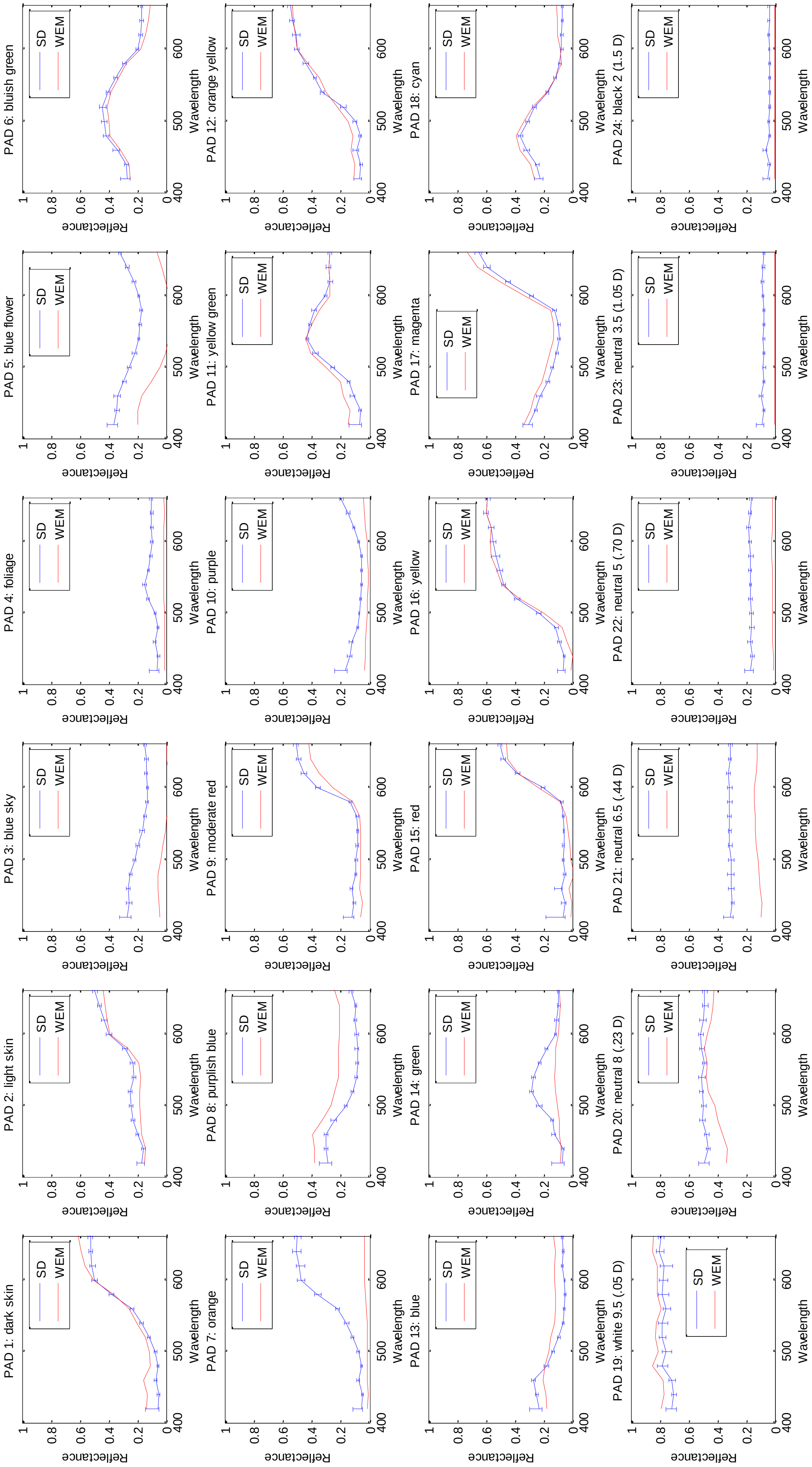


Image 7 : Wiener Estimation Method 10 Terms



	3 Terms Estimation			Thresholds		
Pad	Euclidian Norm	GFC	Spectral Angle	One	Two	Three
1	0,129922	0,995995	0,089527	1	1	0
2	0,121307	0,996934	0,078326	1	1	0
3	0,054225	0,998264	0,05893	1	1	0
4	0,054612	0,991033	0,134021	1	0	0
5	0,078797	0,997547	0,070059	1	1	0
6	0,141737	0,993783	0,111568	1	0	0
7	0,792106	0,989497	0,14506	0	0	0
8	0,125722	0,985718	0,169208	0	0	0
9	0,101643	0,996561	0,082952	1	1	0
10	0,175047	0,961366	0,278875	0	0	0
11	0,119415	0,995372	0,09625	1	1	0
12	0,123186	0,997463	0,071254	1	1	0
13	0,08633	0,988279	0,153258	0	0	0
14	0,136119	0,977982	0,210234	0	0	0
15	0,134064	0,993022	0,1182	1	0	0
16	0,113236	0,999299	0,03745	1	1	1
17	0,114216	0,997833	0,065852	1	1	0
18	0,128469	0,987462	0,15852	0	0	0
19	0,147896	0,999287	0,037754	1	1	1
20	0,094994	0,998879	0,047348	1	1	0
21	0,067888	0,998566	0,053564	1	1	0
22	0,038364	0,998315	0,058054	1	1	0
23	0,026054	0,996746	0,080691	1	1	0
24	0,030116	0,987572	0,157822	0	0	0
Percentage Of Pads With Successful Estimation				70,83%	58,33%	8,33%

	7 Terms Estimation			Thresholds		
Pad	Euclidian Norm	GFC	Spectral Angle	One	Two	Three
1	0,105902	0,997751	0,067081	1	1	0
2	0,25141	0,983197	0,183577	0	0	0
3	0,117618	0,991556	0,130048	1	0	0
4	0,208391	0,980433	0,198148	0	0	0
5	0,207141	0,994626	0,103716	1	0	0
6	0,153266	0,997064	0,076653	1	1	0
7	0,768764	0,942671	0,34025	0	0	0
8	0,102323	0,99038	0,138819	1	0	0
9	0,092122	0,997212	0,074689	1	1	0
10	0,215768	0,947011	0,326999	0	0	0
11	0,14176	0,993169	0,116948	1	0	0
12	0,082404	0,998488	0,055002	1	1	0
13	0,166972	0,959594	0,285242	0	0	0

14	0,181849	0,961391	0,278783	0	0	0
15	0,169986	0,988397	0,152486	0	0	0
16	0,108335	0,999215	0,039638	1	1	1
17	0,097775	0,998388	0,056794	1	1	0
18	0,121625	0,987441	0,158652	0	0	0
19	0,198089	0,999475	0,032395	1	1	1
20	0,215705	0,997648	0,068597	1	1	0
21	0,219022	0,995064	0,099395	1	1	0
22	0,076811	0,993245	0,116296	1	0	0
23	0,246105	0,990115	0,14072	1	0	0
24	0,377675	0,979379	0,203434	0	0	0
Percentage Of Pads With Successful Estimation				62,50%	37,50%	8,33%

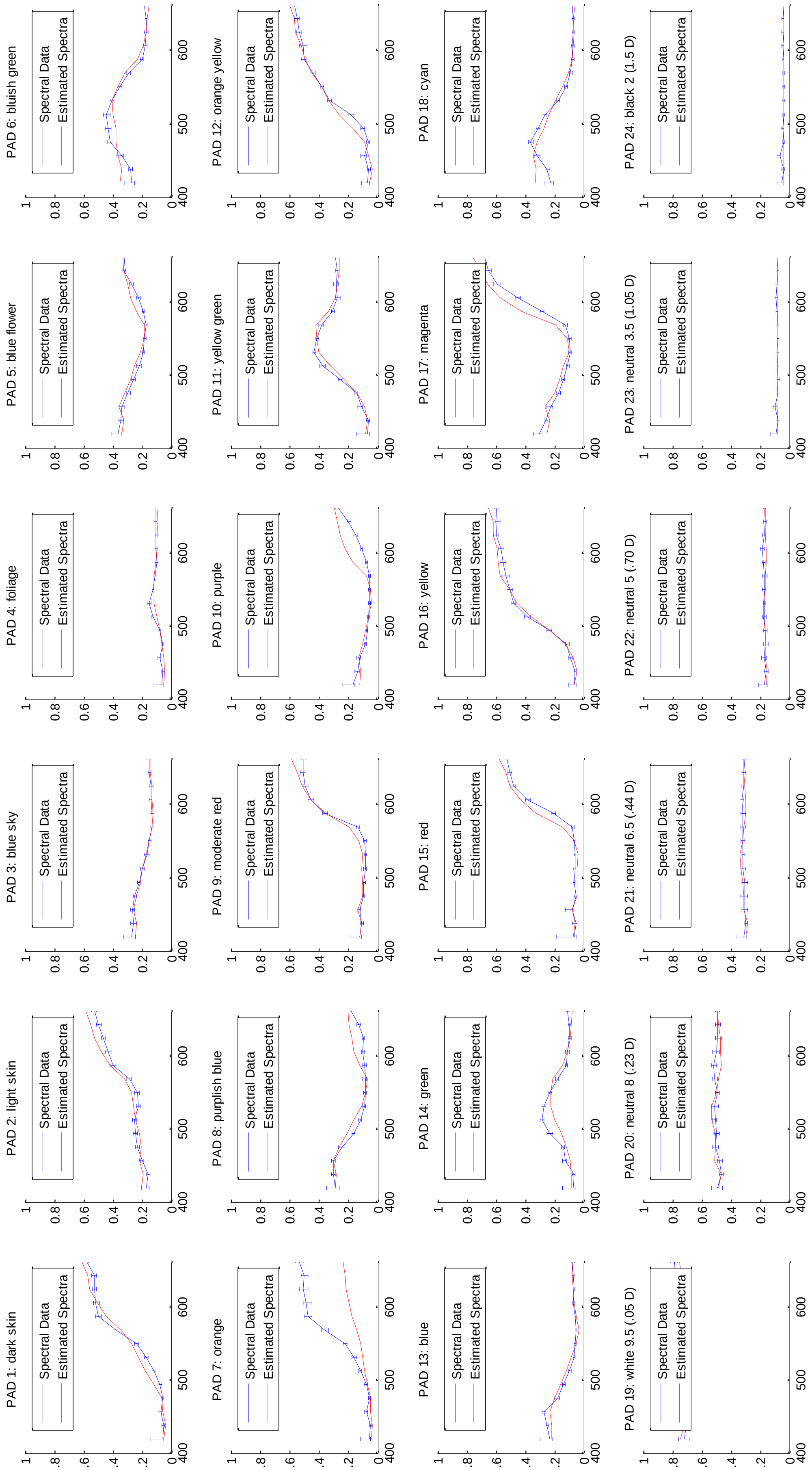
	10 Terms Estimation			Thresholds		
Pad	Euclidian Norm	GFC	Spectral Angle	One	Two	Three
1	0,21936	0,994541	0,104536	1	0	0
2	0,189533	0,997519	0,07046	1	1	0
3	0,665658	0,676056	0,828399	0	0	0
4	0,335904	0,991184	0,132881	1	0	0
5	0,803067	0,786103	0,666317	0	0	0
6	0,149277	0,995679	0,092991	1	1	0
7	1,245597	0,973186	0,232097	0	0	0
8	0,418377	0,976129	0,218934	0	0	0
9	0,246943	0,996384	0,085062	1	1	0
10	0,491529	0,97781	0,211058	0	0	0
11	0,157127	0,99086	0,135305	1	0	0
12	0,122021	0,996728	0,080911	1	1	0
13	0,228628	0,921217	0,3996	0	0	0
14	0,296257	0,950474	0,31604	0	0	0
15	0,16897	0,991269	0,132242	1	0	0
16	0,119596	0,997712	0,067665	1	1	0
17	0,211139	0,999253	0,038656	1	1	1
18	0,124168	0,995656	0,093246	1	1	0
19	0,196928	0,999858	0,016828	1	1	1
20	0,298263	0,994731	0,102698	1	0	0
21	0,71545	0,992088	0,125875	1	0	0
22	0,58334	0,992309	0,124102	1	0	0
23	0,314393	0,993867	0,110804	1	0	0
24	0,160459	0,988328	0,152935	0	0	0
Percentage Of Pads With Successful Estimation				66,67%	33,33%	8,33%

As we can see from the above results, we can have a decent estimation using Wiener. Between the 3, 7 and 10 terms after examining the results provided we saw that the 3 terms estimation can give satisfactory results with a higher success rate in general. So in further use of the algorithm we used the 3 terms version.

The next group of algorithms we checked was the Basis Projection Algorithms without the use of Spectral Sensitivity. In this category we had a number of algorithms described above in the algorithm section. In this algorithm the number of basis we used could significantly change the result. So, we run the algorithms many times and we present the results with the basis number that gave the best fit and Measures Of Spectral Similarities.

The first algorithm that we checked was the Fast Fourier Transform. After running a series of tests we saw that the maximum results (Image 9) came by using 19 basis. If we increase that number we get a better result in the third Threshold percentage but we lose in overall from the other two, which means that we don't get good results in general. A remark that needs to be made is that, with the use of FFT the Estimated spectra contain imaginary parts. In our work we ignored these parts and in fitting and with the Measures Of Spectral Similarities. The total time the FFT used to make the spectral estimation was: 0.007048 seconds (This time is for a specific number of pixels only. Not of an entire spectral cube).

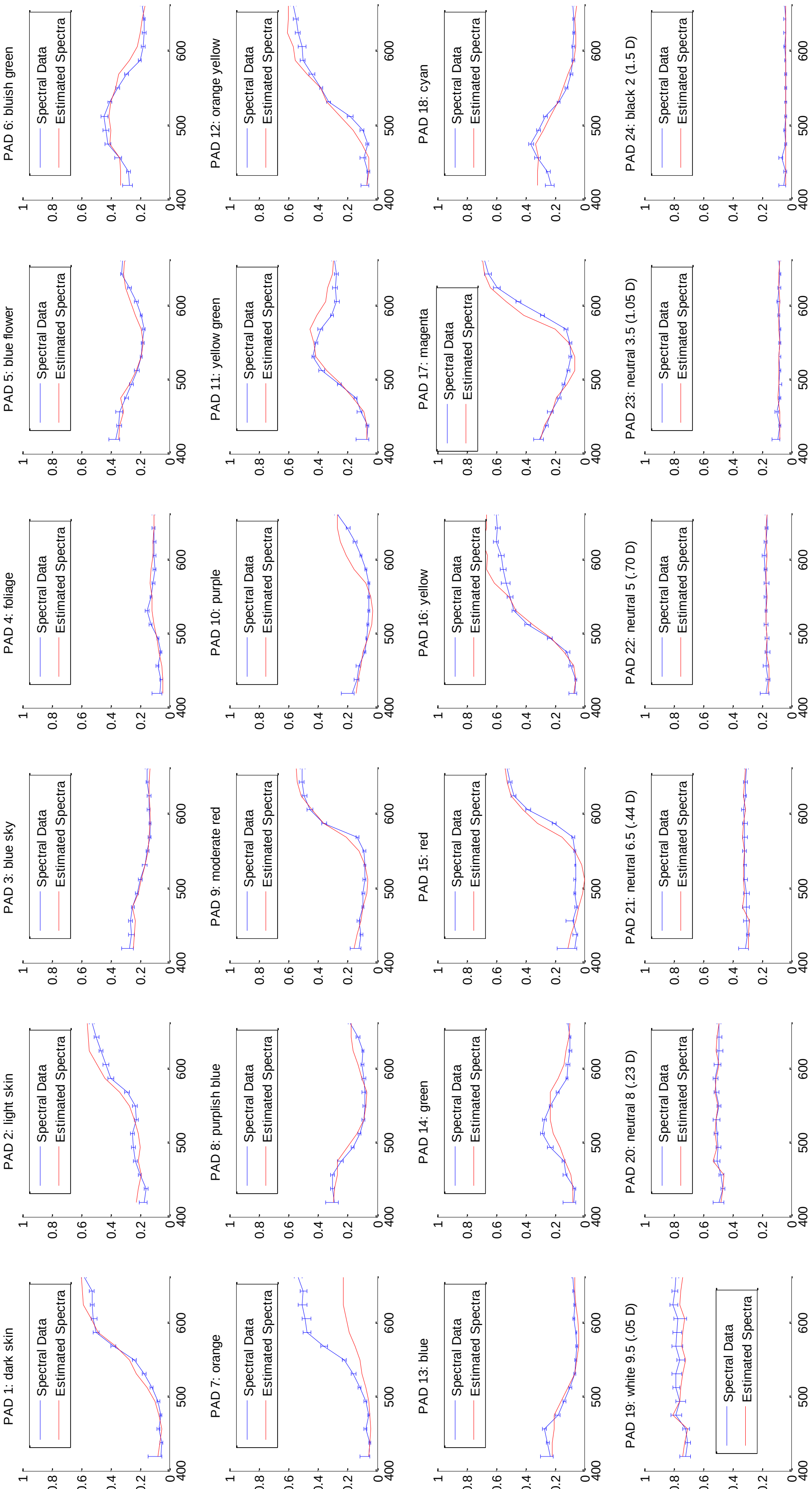
Image 8 : FFT Spectral Estimation



	Fast Fourier Transform Estimation			Thresholds		
Pad	Euclidian Norm	GFC	Spectral Angle	One	Two	Three
1	0,138263	0,996091	0,088446	1	1	0
2	0,147723	0,998155	0,06075	1	1	0
3	0,04106	0,999282	0,03789	1	1	1
4	0,059996	0,99211	0,125703	1	0	0
5	0,106888	0,996801	0,080011	1	1	0
6	0,172678	0,989301	0,146411	0	0	0
7	0,768792	0,990083	0,140953	1	0	0
8	0,144473	0,984385	0,176951	0	0	0
9	0,186225	0,995396	0,095997	1	1	0
10	0,211642	0,959842	0,284359	0	0	0
11	0,122704	0,994465	0,105266	1	0	0
12	0,149304	0,996549	0,083105	1	1	0
13	0,072266	0,991072	0,133728	1	0	0
14	0,160905	0,974883	0,224601	0	0	0
15	0,218357	0,992767	0,120349	1	0	0
16	0,106226	0,999012	0,04445	1	1	1
17	0,297014	0,994139	0,108321	1	0	0
18	0,154727	0,982038	0,189823	0	0	0
19	0,212636	0,998226	0,059579	1	1	0
20	0,090317	0,998877	0,047392	1	1	0
21	0,055986	0,998911	0,046669	1	1	0
22	0,037493	0,998626	0,052418	1	1	0
23	0,02396	0,997473	0,071102	1	1	0
24	0,028924	0,989837	0,142689	0	0	0
Percentage Of Pads With Successful Estimation				75,00%	50,00%	8,33%

Next on the list was Principal Component Analysis Algorithm. As with fft we had a number of basis we could use. In PCA we preferred to use 3 basis since we used three channels for the estimation (R, G, and B). So next we compared the results of PCA with both Wiener and fft. From the results provided later on we saw that for the PCA we used 3 basis. After tests, we noticed that we can use up to 6 basis and still get the same results. If we use less than 3 or more than 6 basis the results worsen dramatically. So, we used 3-6 basis to get our results.

Image 9 : PCA Spectral Estimation

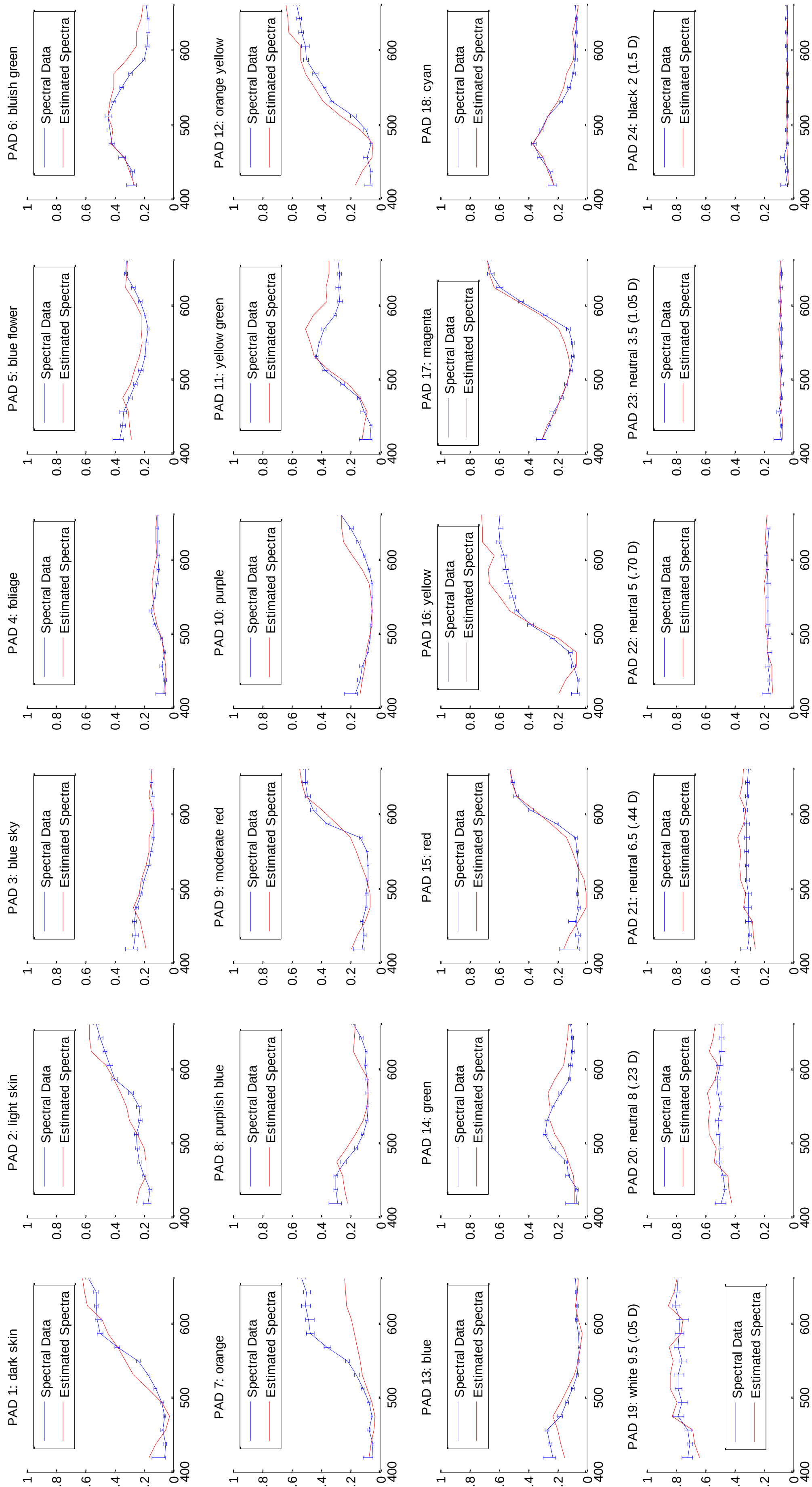


	Principal Component Analysis Estimation			Thresholds		
Pad	Euclidian Norm	GFC	Spectral Angle	One	Two	Three
1	0,12577	0,997114	0,075993	1	1	0
2	0,177728	0,994589	0,104079	1	0	0
3	0,059673	0,998602	0,052892	1	1	0
4	0,062953	0,987056	0,161073	0	0	0
5	0,086049	0,996846	0,07944	1	1	0
6	0,153724	0,992011	0,126487	1	0	0
7	0,762899	0,992201	0,124976	1	0	0
8	0,133332	0,983301	0,183005	0	0	0
9	0,122952	0,996723	0,080976	1	1	0
10	0,20149	0,951343	0,31323	0	0	0
11	0,166101	0,992058	0,126116	1	0	0
12	0,173077	0,998019	0,062948	1	1	0
13	0,094811	0,986541	0,164252	0	0	0
14	0,146475	0,973958	0,228717	0	0	0
15	0,188649	0,98911	0,147717	0	0	0
16	0,226051	0,997508	0,070617	1	1	0
17	0,18844	0,995051	0,099531	1	1	0
18	0,141809	0,983665	0,180994	0	0	0
19	0,157209	0,999242	0,038951	1	1	1
20	0,058468	0,999608	0,028014	1	1	1
21	0,050058	0,999264	0,038381	1	1	1
22	0,034998	0,998624	0,052475	1	1	0
23	0,028322	0,996179	0,087444	1	1	0
24	0,031845	0,985618	0,169803	0	0	0
Percentage Of Pads With Successful Estimation				66,67%	50,00%	12,50%

As we can see from the results of the algorithm, the percentages of the estimation metrics were lower than the other two algorithms. We could see that we can still got a fair estimation on some pads and that on certain pads PCA has the same problems (ex. PAD 7) as Wiener and FFT. But, in certain cases the previous algorithms perform better and give better fit (ex. PAD 16)

The next algorithm we checked was Wavelets. In this case, we tested a number of basis number and we got the same results. So, we chose a default number of 8 basis to use and we present the results (Image 10).

Image 10 : Wavelets Spectral Estimation



	Wavelets Estimation			Thresholds		
Pad	Euclidian Norm	GFC	Spectral Angle	One	Two	Three
1	0,230575	0,990552	0,137574	1	0	0
2	0,231973	0,993335	0,115523	1	0	0
3	0,130559	0,984217	0,177902	0	0	0
4	0,071822	0,986236	0,166107	0	0	0
5	0,1732	0,988465	0,152033	0	0	0
6	0,230563	0,988176	0,153933	0	0	0
7	0,746386	0,984428	0,176709	0	0	0
8	0,179216	0,969745	0,246614	0	0	0
9	0,195057	0,98894	0,148868	0	0	0
10	0,167568	0,965657	0,262836	0	0	0
11	0,287631	0,987959	0,155338	0	0	0
12	0,281518	0,995765	0,092061	1	1	0
13	0,158059	0,956713	0,295307	0	0	0
14	0,189996	0,959075	0,287078	0	0	0
15	0,171175	0,989395	0,145766	0	0	0
16	0,38383	0,994607	0,1039	1	0	0
17	0,123462	0,997985	0,063501	1	1	0
18	0,091222	0,993731	0,112034	1	0	0
19	0,230864	0,997725	0,067469	1	1	0
20	0,287877	0,993467	0,114366	1	0	0
21	0,191168	0,992737	0,120594	1	0	0
22	0,092548	0,992552	0,122128	1	0	0
23	0,050381	0,989783	0,143067	0	0	0
24	0,035528	0,975692	0,22094	0	0	0
Percentage Of Pads With Successful Estimation				41,67%	12,50%	0,00%

So, we saw that we had very bad results for the estimation. We can see from the Measures Of Spectral Similarities that all the pads couldn't pass the third threshold and very few passed the second one. Also we can see from the plot (Image 10) that all of the estimated spectra were a bit wavy. That's because, when using wavelets the results we get have a kind of pulse. To get good result from them we must fit and smooth them. So, wavelets are not good for our work, since we don't get results and after-acquisition processing is required in order to make to get better estimates.

Following, we tested the Singular Value Decomposition (SVD) algorithm. In this algorithm we used a number of bases between 3-6. Lower or higher than that, results were 0% for all Thresholds.

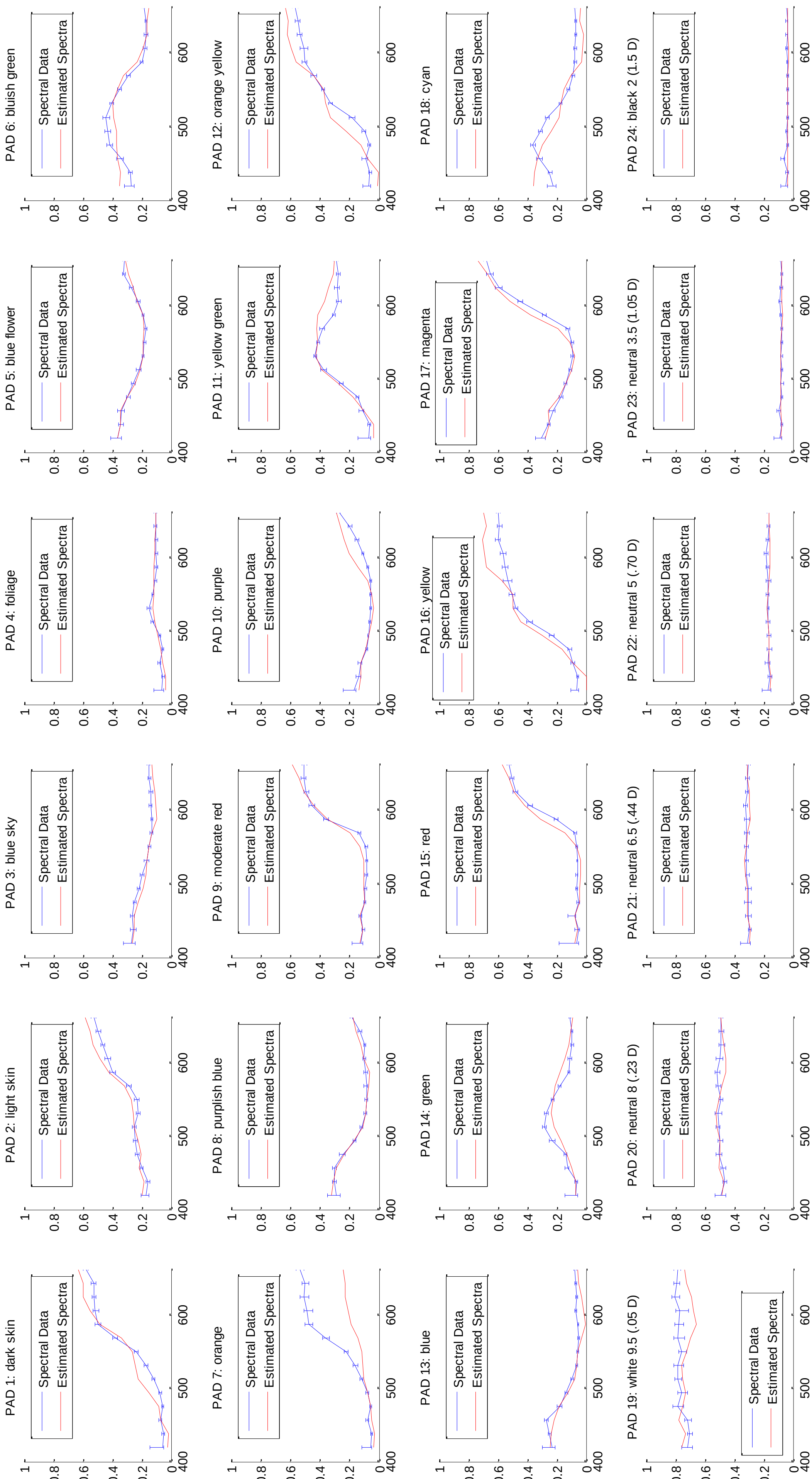
Image 11 : SVD Spectral Estimation

	SVD Estimation			Thresholds		
Pad	Euclidian Norm	GFC	Spectral Angle	One	Two	Three
1	0,12843	0,996984	0,077685	1	1	0
2	0,152394	0,996693	0,081348	1	1	0
3	0,058641	0,998685	0,051291	1	1	0
4	0,055913	0,989889	0,142325	1	0	0
5	0,068709	0,99803	0,062772	1	1	0
6	0,1252	0,994815	0,101882	1	0	0
7	0,763302	0,991469	0,130718	1	1	0
8	0,116007	0,987374	0,159079	0	0	0
9	0,12402	0,996654	0,081832	1	1	0
10	0,18344	0,960175	0,283168	0	0	0
11	0,148556	0,994188	0,107863	1	1	0
12	0,176396	0,997864	0,065367	1	1	0
13	0,094176	0,986725	0,163125	1	0	0
14	0,135838	0,977851	0,210864	0	0	0
15	0,16827	0,991828	0,127932	1	0	0
16	0,212318	0,998496	0,054857	1	1	1
17	0,160449	0,99705	0,076828	1	1	0
18	0,132214	0,985862	0,168354	0	0	0
19	0,161784	0,999133	0,041649	1	1	1
20	0,095756	0,998834	0,04829	1	1	0
21	0,068314	0,998533	0,054171	1	1	0
22	0,038931	0,998268	0,058857	1	1	0
23	0,026432	0,996685	0,081448	1	1	0
24	0,03029	0,987667	0,157216	1	0	0
Percentage Of Pads With Successful Estimation				70,83%	54,17%	4,17%

From the results above we can see that SVD gives quite good results. In comparison with fft and wiener we saw that we fot approximately the same percentages for the thresholds with SVD coming third. From the plots (image 11) we can see that we have again about the same results. We can spot the same difficulties (ex. Pad 7) or the same good fit (Pad 17). But, in general FFT and WEM give a little better results.

Next was Discrete Cosine Transform (DCT) algorithm. In this case we got our best results using 15 basis.

Image 12 : DCT Estimation

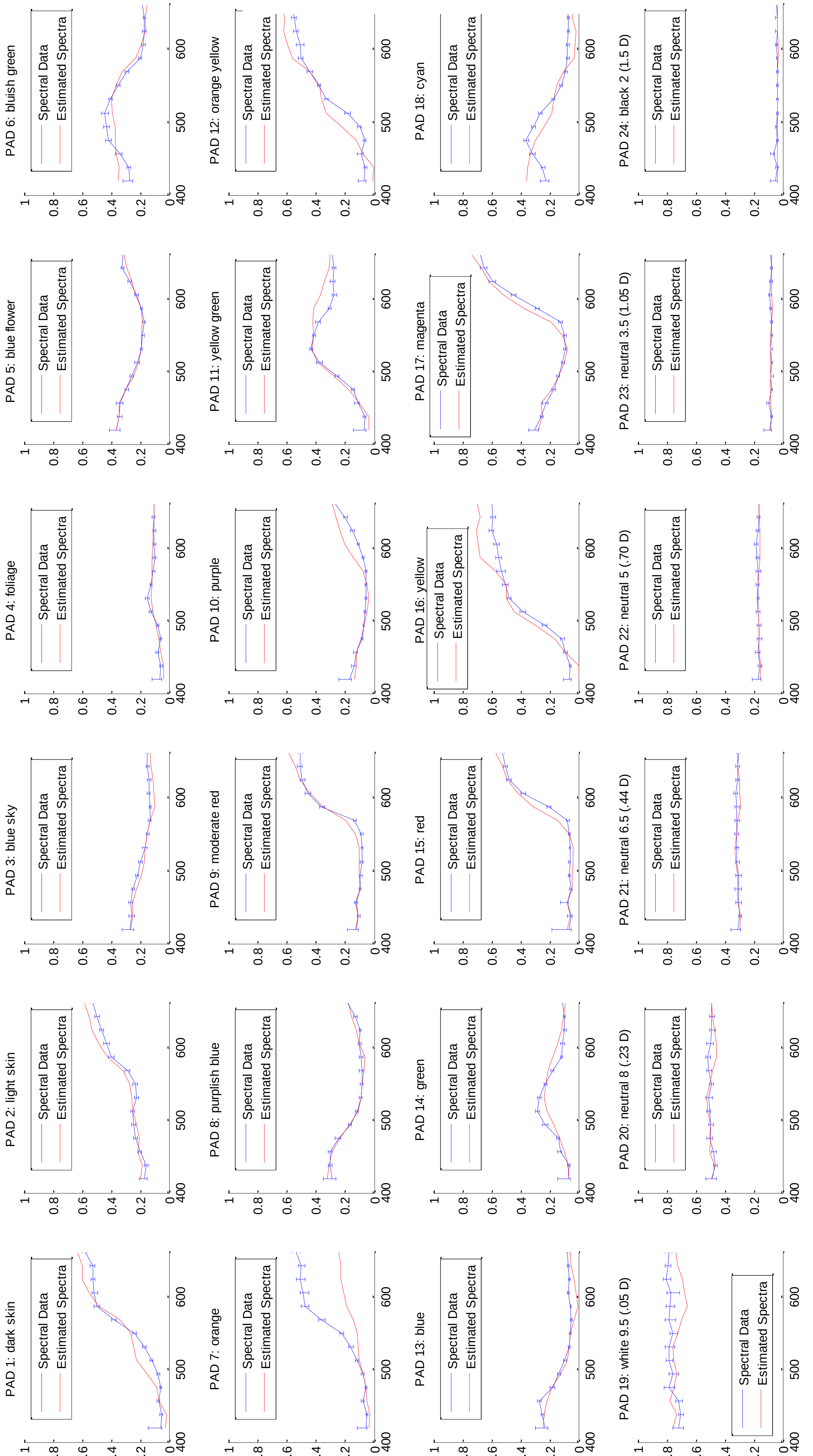


	DCT Estimation			Thresholds		
Pad	Euclidian Norm	GFC	Spectral Angle	One	Two	Three
1	0,218885	0,99378	0,111595	1	0	0
2	0,15265	0,998494	0,054881	1	1	0
3	0,086881	0,996787	0,080184	1	1	0
4	0,056129	0,989645	0,144036	0	0	0
5	0,042153	0,999244	0,038887	1	1	1
6	0,182203	0,988058	0,154697	0	0	0
7	0,742602	0,988923	0,148977	0	0	0
8	0,109799	0,988541	0,151534	0	0	0
9	0,174934	0,995102	0,099015	1	1	0
10	0,169905	0,968083	0,253331	0	0	0
11	0,175425	0,99316	0,117032	1	0	0
12	0,310489	0,991317	0,131879	1	0	0
13	0,109961	0,981149	0,194478	0	0	0
14	0,130324	0,979964	0,200513	0	0	0
15	0,198751	0,993059	0,117891	1	0	0
16	0,323212	0,996356	0,085398	1	1	0
17	0,196456	0,996581	0,08272	1	1	0
18	0,248069	0,948574	0,322097	0	0	0
19	0,259863	0,997572	0,069701	1	1	0
20	0,090664	0,998879	0,047346	1	1	0
21	0,053076	0,999016	0,044366	1	1	1
22	0,037941	0,998705	0,050902	1	1	0
23	0,025464	0,997436	0,071623	1	1	0
24	0,030387	0,989878	0,142402	0	0	0
Percentage Of Pads With Successful Estimation				62,50%	45,83%	8,33%

In this case we can see approximately the same thing as with SVD. We about the same results with SVD and we can see from the plots that we have bad estimation in the same pads as before. So, we can assume that we get good results but still not as good as Wiener and FFT.

Next, and last algorithm we checked was Hilbert Transformation. After examining we concluded in using 3 basis to get the best results possible from this algorithm.

Image 13: Hilbert Transformation Estimation



	Hilbert Tranform Estimation			Thresholds		
Pad	Euclidian Norm	GFC	Spectral Angle	One	Two	Three
1	0,219957	0,988259	0,15339	0	0	0
2	0,186528	0,995257	0,097438	1	1	0
3	0,112649	0,988601	0,151132	0	0	0
4	0,057169	0,989527	0,144857	0	0	0
5	0,196619	0,989451	0,145379	0	0	0
6	0,224539	0,985956	0,167792	0	0	0
7	0,786018	0,981637	0,191936	0	0	0
8	0,224963	0,962418	0,275027	0	0	0
9	0,205597	0,988346	0,152816	0	0	0
10	0,201042	0,960141	0,283292	0	0	0
11	0,152764	0,991378	0,131412	1	0	0
12	0,161911	0,995098	0,09906	1	1	0
13	0,142718	0,963767	0,270014	0	0	0
14	0,173769	0,962583	0,274418	0	0	0
15	0,117426	0,996629	0,082135	1	1	0
16	0,107081	0,999142	0,041431	1	1	1
17	0,223151	0,996141	0,087884	1	1	0
18	0,164255	0,981787	0,191146	0	0	0
19	0,115766	0,999503	0,031531	1	1	1
20	0,199959	0,997067	0,076606	1	1	0
21	0,134422	0,996443	0,084367	1	1	0
22	0,064805	0,996027	0,089165	1	1	0
23	0,039757	0,993303	0,115797	1	0	0
24	0,032627	0,980137	0,199647	0	0	0
Percentage Of Pads With Successful Estimation				45,83%	37,50%	8,33%

In this case we can see that we don't get as good results as before. In the plots (Image 13) we can notice that Hilbert cannot get a good fit in many pads. Also we see that we get low percentages on our threshold rate which means that Hilbert doesn't perform as good as Wiener and FFT.

Next step was to check which algorithm was the best (as it concerns the Measures Of Spectral Similarities: GFC, Spectral Angle, Euclidian Norm). After completing all tests needed for those algorithms we concluded that the best algorithms to use (with little result difference between them) was the Wiener Estimation Method and the Basis Projection Algorithm Fast Fourier Transform.

	Table 1	Thresholds		
No	Algorithm	One	Two	Three
1	Wiener	75,00%	50,00%	8,33%
2	FFT	70,83%	58,33%	8,33%
3	PCA	66,67%	50,00%	12,50%
4	Wavelets	41,67%	12,50%	0%
5	SVD	70,83%	54.17%	4,17%
	Hilbert Trans.	45,83%	37,50%	8,33%
6	DCT	65,50%	45,83%	8,33%

Finally, the last criterion that needed to be met was time. After running a number of tests with various data we concluded that the best algorithms to be used were Wiener Estimation Method and FFT.

No	Algorithm	Time
1	Wiener	0.004710 s
2	FFT	0.006631 s
3	PCA	0.016213 s
4	Wavelets	0.120807 s
5	SVD	0.162944 s
	Hilbert	0.012850 s
6	DCT	0.012850 s

From all the above statistics and plots we can see that we can validate the algorithms and get a spectra estimate by using only the RGB value of each pad as input. After validating the use of the algorithms we tested the performance of the two best algorithms by using different kinds of input.

4.2 Narrow Bands

After validating the use of the algorithms the next logical step was to see if we can improve the accuracy of them. The first thought we had, was to use Narrow Band filters along with the RGB camera.

We had the option of two different sets of Narrow band filters.

- 464 nm
- 542 nm
- 639 nm

And

- 422 nm
- 503 nm
- 572 nm

As mentioned before, because we prefer the use of MUSIS HS. We can use the bands of MUSIS that are near the Narrow Band Filters described above. So we had the following triple bands filters:

- 460 nm
- 540 nm
- 640 nm

And

- 420 nm
- 500 nm
- 580 nm

The main question that remained to be answered was whether the use of Narrow Band filters will give us better results than the use of RGB values only. So, since we found two algorithms that are fit for job in the previous section , the results of the algorithm will be shown and compared. First of all we checked the Wiener Estimation Algorithm. As we mentioned before we used only the 3 terms estimation since the more terms don’t actually give better results. So, next we provide the graphs and the tables of the estimated spectra using a Triple band filter instead of RGB.

Image 14 : Wiener Estimation With NB 1

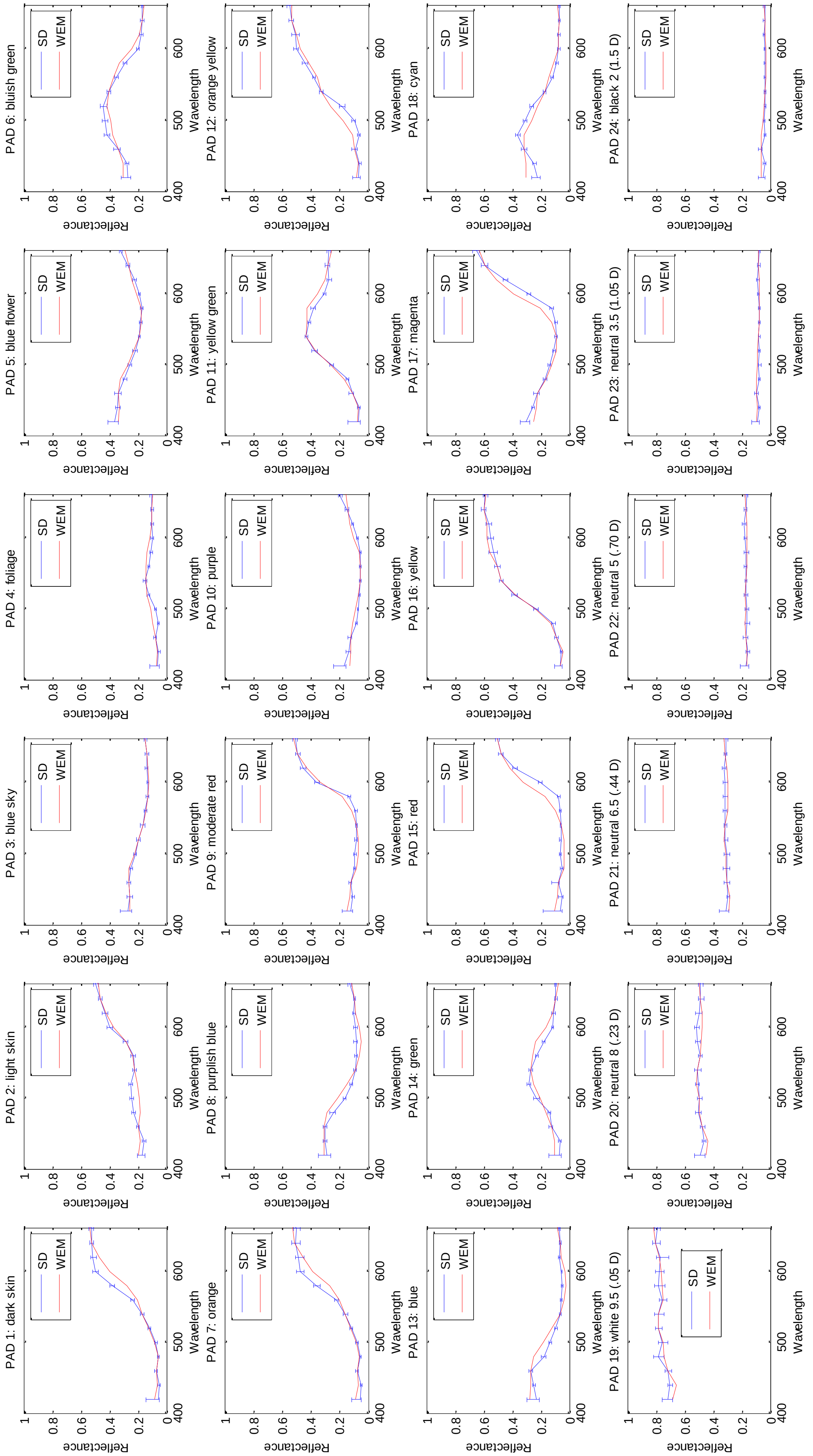
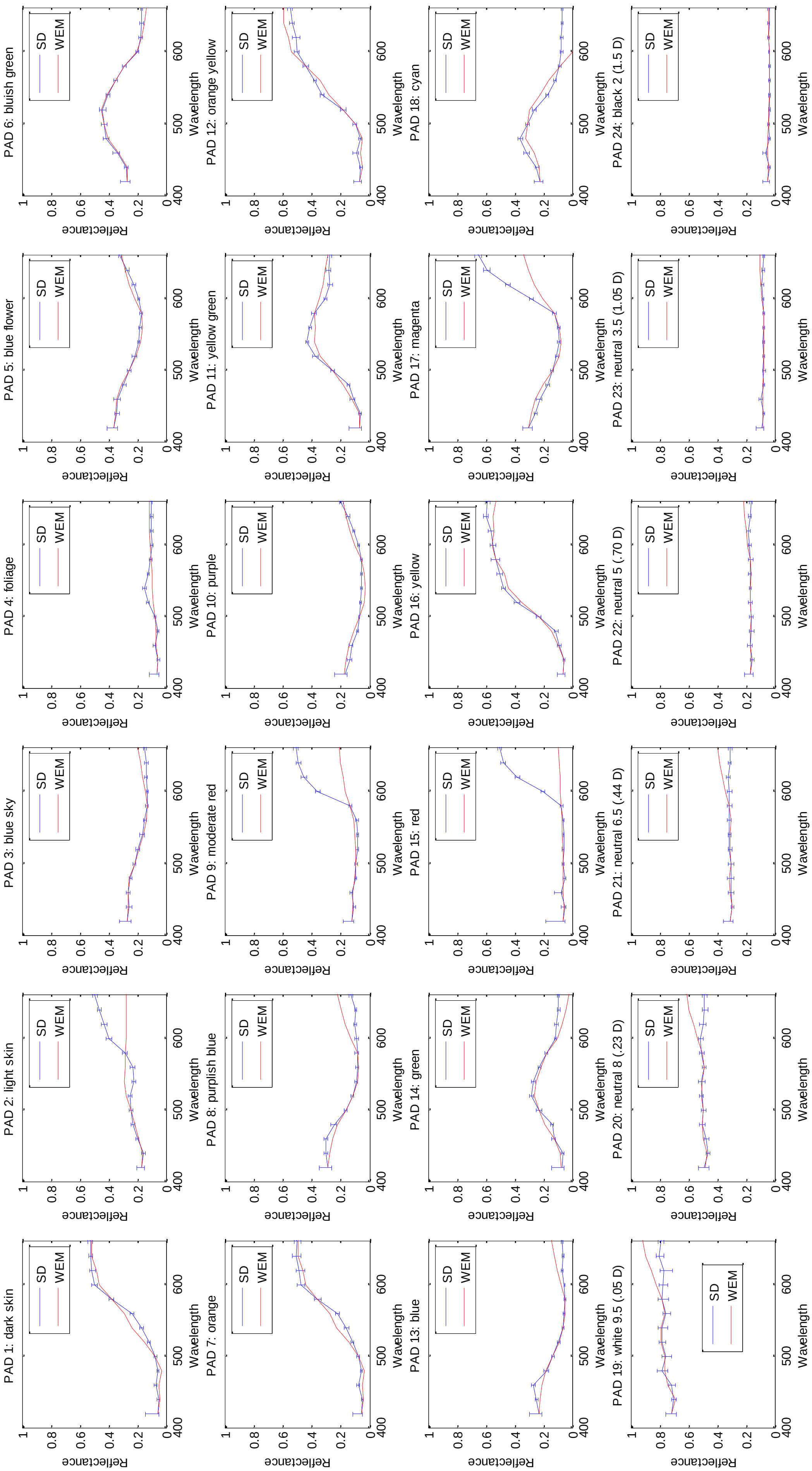


Image 15: Wiener Estimation NB2



	Wiener Estimation NB 1			Thresholds		
Pad	Euclidian Norm	GFC	Spectral Angle	One	Two	Three
1	0,165189	0,995755	0,092174	1	1	0
2	0,132454	0,996738	0,080791	1	1	0
3	0,027814	0,999292	0,037629	1	1	1
4	0,066191	0,990382	0,138803	1	0	0
5	0,069864	0,997838	0,065764	1	1	0
6	0,116302	0,99514	0,098629	1	1	0
7	0,139146	0,995115	0,09888	1	1	0
8	0,164173	0,974263	0,227369	0	0	0
9	0,096551	0,997205	0,074779	1	1	0
10	0,226133	0,947359	0,325913	0	0	0
11	0,099866	0,996191	0,087308	1	1	0
12	0,131406	0,996156	0,087712	1	1	0
13	0,114501	0,988411	0,152389	0	0	0
14	0,115875	0,984234	0,177805	0	0	0
15	0,173038	0,990495	0,137985	1	0	0
16	0,045797	0,999695	0,024704	1	1	1
17	0,163737	0,994361	0,10625	1	0	0
18	0,12948	0,986004	0,167506	0	0	0
19	0,107967	0,999345	0,036198	1	1	1
20	0,103197	0,998533	0,05417	1	1	0
21	0,070802	0,998256	0,059062	1	1	0
22	0,04464	0,997857	0,065481	1	1	0
23	0,035019	0,995932	0,090233	1	1	0
24	0,041499	0,977302	0,213469	0	0	0
	Percentage Of Pads With Successful Estimation			75,00%	62,50%	12,50%

	Wiener Estimation NB 2			Thresholds		
Pad	Euclidian Norm	GFC	Spectral Angle	One	Two	Three
1	0,134533	0,996311	0,08592	1	1	0
2	0,532111	0,949193	0,320134	0	0	0
3	0,10901	0,992088	0,125873	1	0	0
4	0,077088	0,980212	0,199265	0	0	0
5	0,056888	0,998793	0,049135	1	1	0
6	0,109523	0,995988	0,089609	1	1	0
7	0,111255	0,996817	0,079803	1	1	0
8	0,173462	0,97362	0,230203	0	0	0
9	0,675785	0,944244	0,335506	0	0	0
10	0,190163	0,954771	0,301907	0	0	0
11	0,107539	0,995402	0,095936	1	1	0
12	0,15587	0,997152	0,07549	1	1	0

13	0,162521	0,957589	0,292281	0	0	0
14	0,178821	0,960911	0,280524	0	0	0
15	0,881935	0,83966	0,574139	0	0	0
16	0,127485	0,99894	0,046052	1	1	0
17	0,662304	0,957381	0,293001	0	0	0
18	0,310793	0,916452	0,411674	0	0	0
19	0,292083	0,997395	0,072194	1	1	0
20	0,303504	0,99286	0,119571	1	0	0
21	0,208866	0,991778	0,128324	1	0	0
22	0,114327	0,991818	0,128012	1	0	0
23	0,053983	0,99027	0,139613	1	0	0
24	0,024316	0,989729	0,143449	0	0	0
Percentage Of Pads With Successful Estimation				54,17%	33,33%	0,00%

So after evaluating the above results we got the following table,

	Table 2	Thresholds		
No	Algorithm	One	Two	Three
1	Wiener	75,00%	50,00%	8,33%
2	Wiener NB1	75,00%	62,50%	12,50%
3	Wiener NB 2	54,17%	33,33%	0,00%

So we see that the use of Narrow band filters does not give better results for the estimation in both cases. This seems quite logical since the RGB filter is broadband while the Narrow Band filters are narrow. That tells us that we have more unknown “parts” of the spectrum so less a priori knowledge. We see that NB1 gives better results that the NB2 and that it performs a little better from RGB. The reason that NB1 gives good results is because its narrow bands cover all three major areas of spectrum and give proper information for the estimation.

On the other hand the last band of the second one is 580nm. So, any information beyond that are unknown, thus the bad estimation after 580nm. NB1 last band is at 640nm which is the boundary of visible spectrum so we don’t get any problem from that.

Next we showed the effect of Narrow Band Filters on the FFT.

Image 16 : FFT Estimation Using NB1

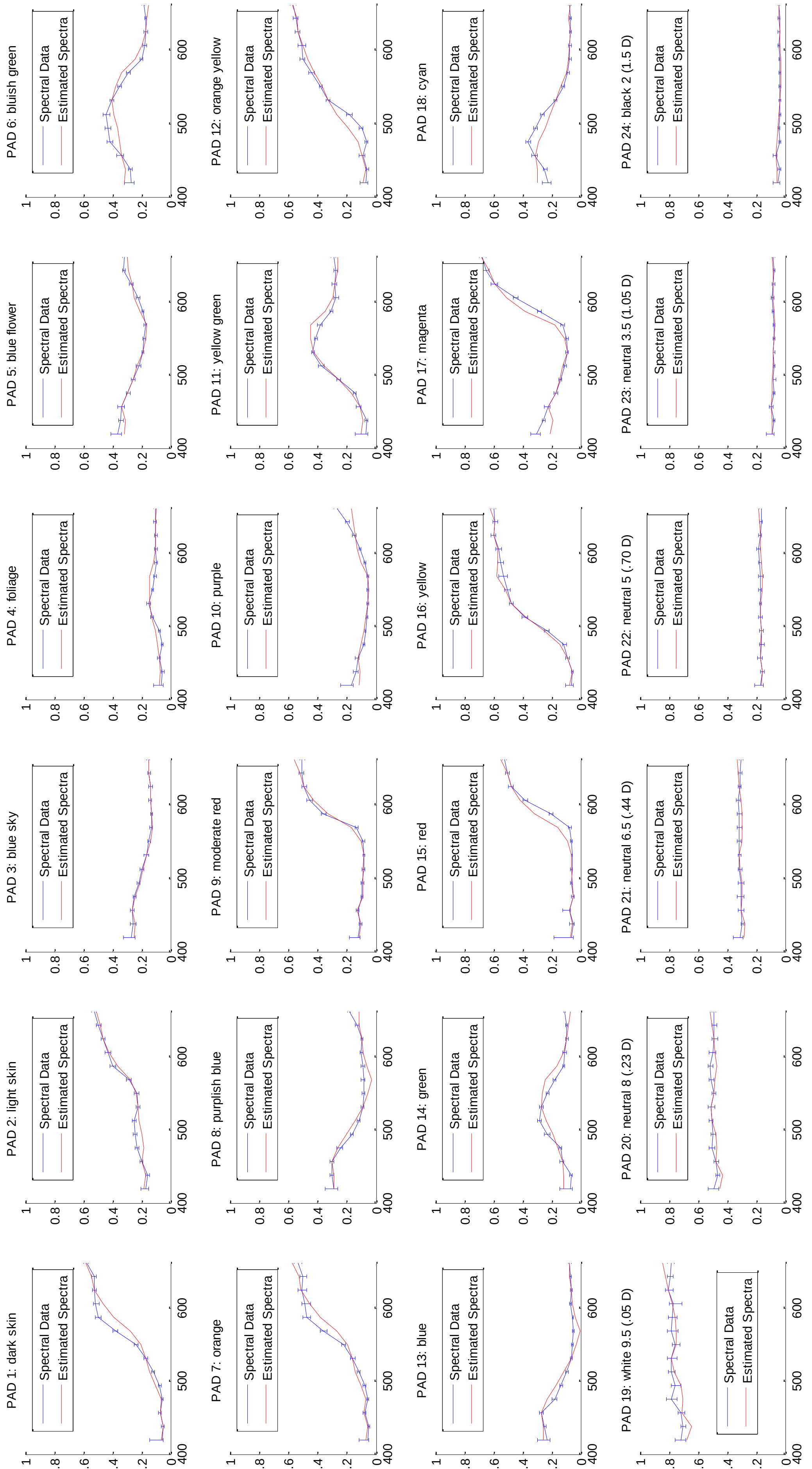
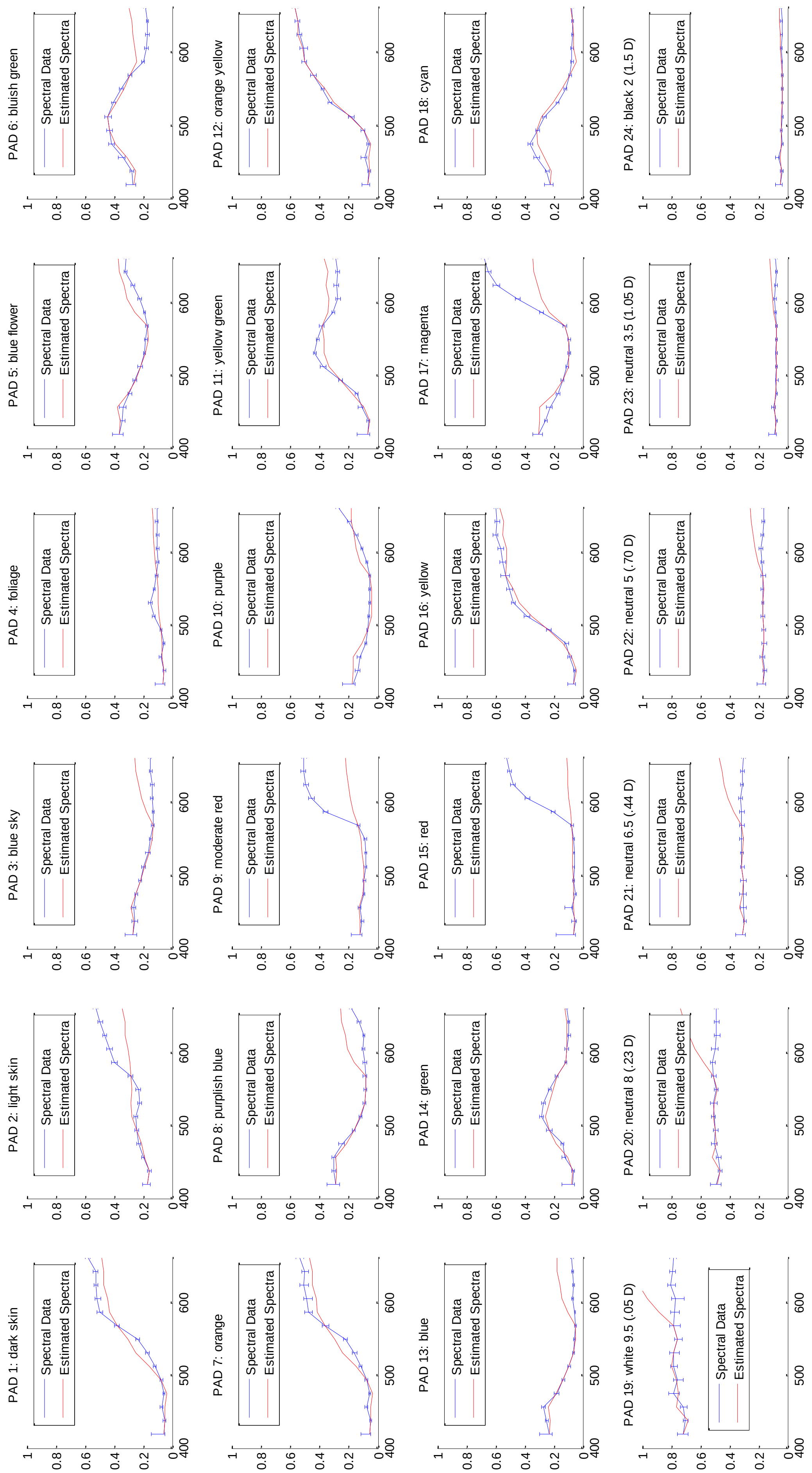


Image 17: FFT Estimation Using NB2



	FFT Estimation NB 1			Thresholds		
Pad	Euclidian Norm	GFC	Spectral Angle	One	Two	Three
1	0,163042	0,994354	0,106312	1	0	0
2	0,097499	0,99829	0,058495	1	1	0
3	0,032683	0,999483	0,032153	1	1	1
4	0,067198	0,989664	0,1439	0	0	0
5	0,078047	0,997598	0,069322	1	1	0
6	0,17543	0,989026	0,148287	0	0	0
7	0,162837	0,992719	0,120748	1	0	0
8	0,16342	0,97505	0,223851	0	0	0
9	0,138095	0,995348	0,096494	1	1	0
10	0,216701	0,95551	0,299411	0	0	0
11	0,132976	0,993309	0,115746	1	0	0
12	0,153702	0,994891	0,101124	1	0	0
13	0,091177	0,989361	0,146001	0	0	0
14	0,153418	0,971885	0,237688	0	0	0
15	0,181797	0,992407	0,123312	1	0	0
16	0,070005	0,99954	0,030328	1	1	1
17	0,195386	0,992832	0,119801	1	0	0
18	0,145778	0,981961	0,190227	0	0	0
19	0,146733	0,998831	0,048361	1	1	0
20	0,109051	0,998433	0,055983	1	1	0
21	0,073037	0,998156	0,060736	1	1	0
22	0,041238	0,998027	0,062828	1	1	0
23	0,027788	0,997117	0,075948	1	1	0
24	0,036083	0,98081	0,196222	0	0	0
	Percentage Of Pads With Successful Estimation			66,67%	41,67%	8,33%

FFT Estimation Triple 2				Thresholds		
Pad	Euclidian Norm	GFC	Spectral Angle	One	Two	Three
1	0,241653	0,990963	0,134544	1	0	0
2	0,410976	0,973942	0,22879	0	0	0
3	0,236303	0,971127	0,240885	0	0	0
4	0,09308	0,972921	0,233247	0	0	0
5	0,164235	0,994651	0,103479	1	0	0
6	0,251522	0,980778	0,196387	0	0	0
7	0,204229	0,992098	0,125793	1	0	0
8	0,232248	0,962169	0,275943	0	0	0
9	0,666005	0,943452	0,337901	0	0	0
10	0,212837	0,938452	0,352675	0	0	0
11	0,19019	0,987995	0,155107	0	0	0
12	0,053148	0,999371	0,035457	1	1	1
13	0,223423	0,935478	0,361188	0	0	0
14	0,083941	0,991544	0,130136	1	0	0
15	0,859154	0,863825	0,527982	0	0	0
16	0,110214	0,999494	0,031817	1	1	1
17	0,668085	0,947207	0,326387	0	0	0
18	0,105606	0,990946	0,134664	1	0	0
19	0,653055	0,988655	0,150772	0	0	0
20	0,529354	0,981689	0,191662	0	0	0
21	0,342343	0,981358	0,193394	0	0	0
22	0,183714	0,98233	0,188268	0	0	0
23	0,084659	0,981707	0,191568	0	0	0
24	0,038176	0,983422	0,182342	0	0	0
Percentage Of Pads With Successful Estimation				29,17%	8,33%	8,33%

So after evaluating the above results we get the following table,

	Table 3	**Thresholds**		
No	Algorithm	One	Two	Three
1	FFT	70,83%	58,33%	8,33%
2	FFT NB 1	66,67%	41,67%	8,33%
3	FFT NB 2	29,17%	8,33%	8,33%

From all the above tables we can see that for we don't get good results using the Triple Band Filters on FFT. As with Wiener we can see that we get the worst results by using the NB2 filter since we don't have any information beyond the point of 580nm.

	Table 3	Thresholds		
No	Algorithm	One	Two	Three
1	Wiener NB1	75,00%	62,50%	12,50%
1	FFT NB 1	66,67%	41,67%	8,33%
2	Wiener NB 2	54,17%	33,33%	0,00%
2	FFT NB 2	29,17%	8,33%	8,33%

So, we concluded from the results (Table 3) that the use of only Narrow Band filters does not provide better estimation over RGB. In addition we can see the superiority of the Wiener algorithm over FFT and also that between the two sets of Narrow Band Filters we get the best results from NB1..

4.3 RGB Broadband and Narrow Bands

After concluding that the use of Narrow Band filters doesn't give better estimation of spectra over RGB, we came up with the idea to mix the 3 Broadband RGB and the Narrow Band Filters. So we will have six bands as input. Three from RGB and three Narrow Bands.

The importance of this idea is that we want to cover all possible areas of data with the use of six bands. So, we get good spectral information with the use of 3 Narrow Bands and we complete any spectra information needed by using three Broadband RGB filters. As, we have seen for example, NB2 cannot make a very accurate estimation beyond the point of 580 nm since we don't have any information after that point. Same with NB1 were we don't get information after 640 nm. But as we said recent studies showed that the visible spectrum is from 420 nm to 660-670 nm, so we don't have a major data problem with NB1.

So, the only thing that changes in our algorithms is that instead of using a 24x3 matrix which contains the RGB values of all pads, we use a 24x6 matrix which contains both the RGB values and the Narrow Band values of all pads. Again we checked this input for both FFT and Wiener Algorithms.

Image 18 : WEM Using RGB & NB1

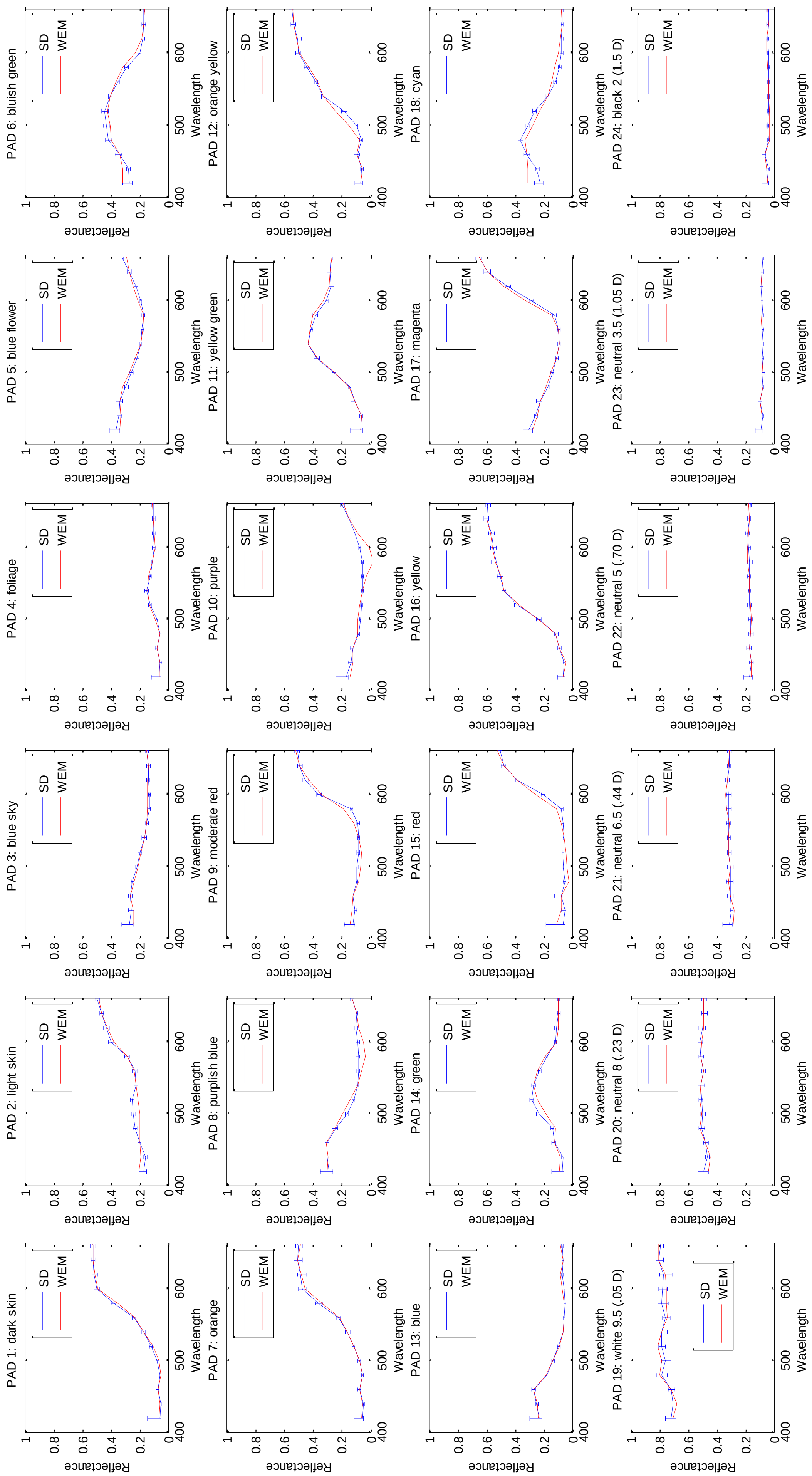
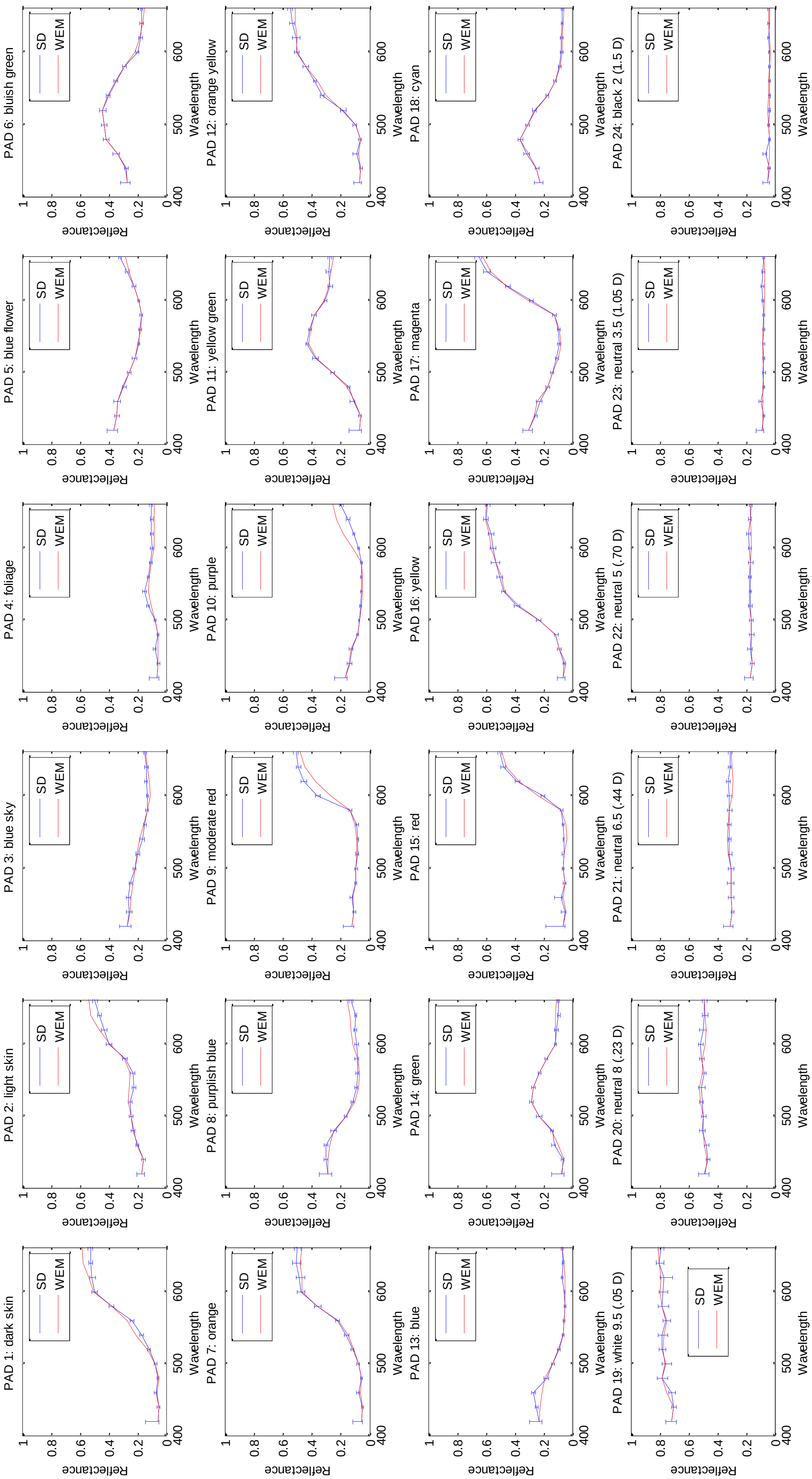


Image 19 : WEM Using RGB & NB2



	Wiener Estimation Triple 1 & RGB			Thresholds		
Pad	Euclidian Norm	GFC	Spectral Angle	One	Two	Three
1	0,091039	0,998847	0,048021	1	1	0
2	0,122941	0,996942	0,078229	1	1	0
3	0,047928	0,998006	0,063157	1	1	0
4	0,035601	0,996812	0,079868	1	1	0
5	0,060504	0,998384	0,056865	1	1	0
6	0,104354	0,996101	0,088335	1	1	0
7	0,099249	0,998652	0,051924	1	1	0
8	0,118398	0,98695	0,161729	0	0	0
9	0,100549	0,997106	0,076096	1	1	0
10	0,17122	0,973186	0,232097	0	0	0
11	0,049594	0,99903	0,044056	1	1	1
12	0,092062	0,998181	0,060319	1	1	0
13	0,050275	0,996191	0,087308	1	1	0
14	0,0752	0,993532	0,113801	1	0	0
15	0,119711	0,996467	0,084084	1	1	0
16	0,032451	0,999872	0,015972	1	1	1
17	0,066913	0,999169	0,040778	1	1	1
18	0,129559	0,98658	0,164015	0	0	0
19	0,073525	0,999696	0,024659	1	1	1
20	0,070859	0,999322	0,036829	1	1	1
21	0,056812	0,998969	0,045402	1	1	0
22	0,034411	0,998824	0,048508	1	1	0
23	0,023625	0,998374	0,057042	1	1	0
24	0,031727	0,984388	0,176935	0	0	0
	Percentage Of Pads With Successful Estimation			83,33%	79,17%	20,83%

	Wiener Estimation Triple 2 & RGB			Thresholds		
Pad	Euclidian Norm	GFC	Spectral Angle	One	Two	Three
1	0,101505	0,998863	0,047701	1	1	0
2	0,096733	0,998727	0,050472	1	1	0
3	0,042088	0,998511	0,054576	1	1	0
4	0,068643	0,994716	0,102847	1	0	0
5	0,056546	0,99883	0,04838	1	1	0
6	0,080003	0,997817	0,066082	1	1	0
7	0,083889	0,999491	0,031915	1	1	1
8	0,112804	0,988323	0,152968	0	0	0
9	0,129355	0,995922	0,090338	1	1	0
10	0,148049	0,972236	0,23619	0	0	0

11	0,082544	0,997856	0,065489	1	1	0
12	0,066813	0,999565	0,029513	1	1	1
13	0,065219	0,994613	0,103845	1	0	0
14	0,043162	0,997727	0,067434	1	1	0
15	0,067712	0,998181	0,060323	1	1	0
16	0,035581	0,999822	0,018845	1	1	1
17	0,062017	0,999345	0,036209	1	1	1
18	0,030666	0,999277	0,038037	1	1	1
19	0,08728	0,999691	0,02486	1	1	1
20	0,07711	0,999225	0,039373	1	1	1
21	0,051094	0,999097	0,04249	1	1	1
22	0,034417	0,998675	0,05149	1	1	0
23	0,028196	0,996931	0,078364	1	1	0
24	0,028909	0,983937	0,179477	0	0	0
	Percentage Of Pads With Successful Estimation			87,50%	79,17%	33,33%

So after evaluating the above results we get the following table,

	Table 4	**Thresholds**		
No	Algorithm	One	Two	Three
1	Wiener	75,00%	50,00%	8,33%
2	Wiener NB 1 & RGB	83,33%	79,17%	20,83%
3	Wiener NB 2 & RGB	87,50%	79,17%	33,33%

As we can see from the results in Table 4, we got very good results over the previous input bands. The percentages rise with the use of both sets if Narrow Band filters. Its noticeable that this time NB2 (in combination with RGB) gave better results that NB1 (in combination with RGB). We can see that in both cases we have a very good fit and very good performance with our Measures Of Spectral Similarities. We saw that although NB2, as we mentioned before, doesn't have information beyond 580nm, with the help of the 3 Broadband RGB bands gets the information needed and performs very well. From the performance tables we could also see that we got different performance for some pads from those two input methods. That's because the two set of bands focus on certain nm. So, if a pad has a deep or peek on a certain wavelength the method that has this wavelength on the Narrow Bands will perform better.

Now the FFT estimation algorithm was checked with this input.

Image 20 : FFT Using NB1 & RGB

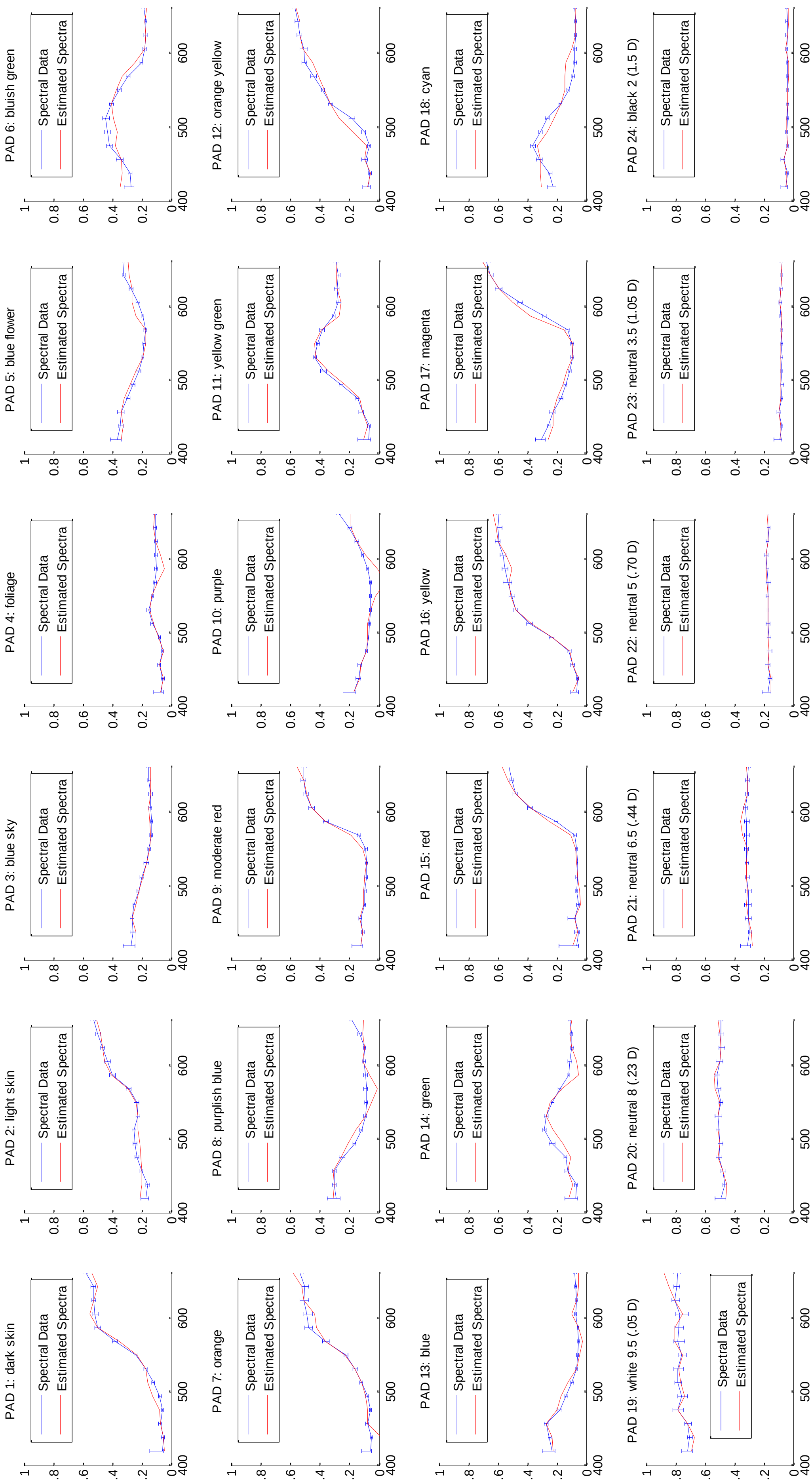
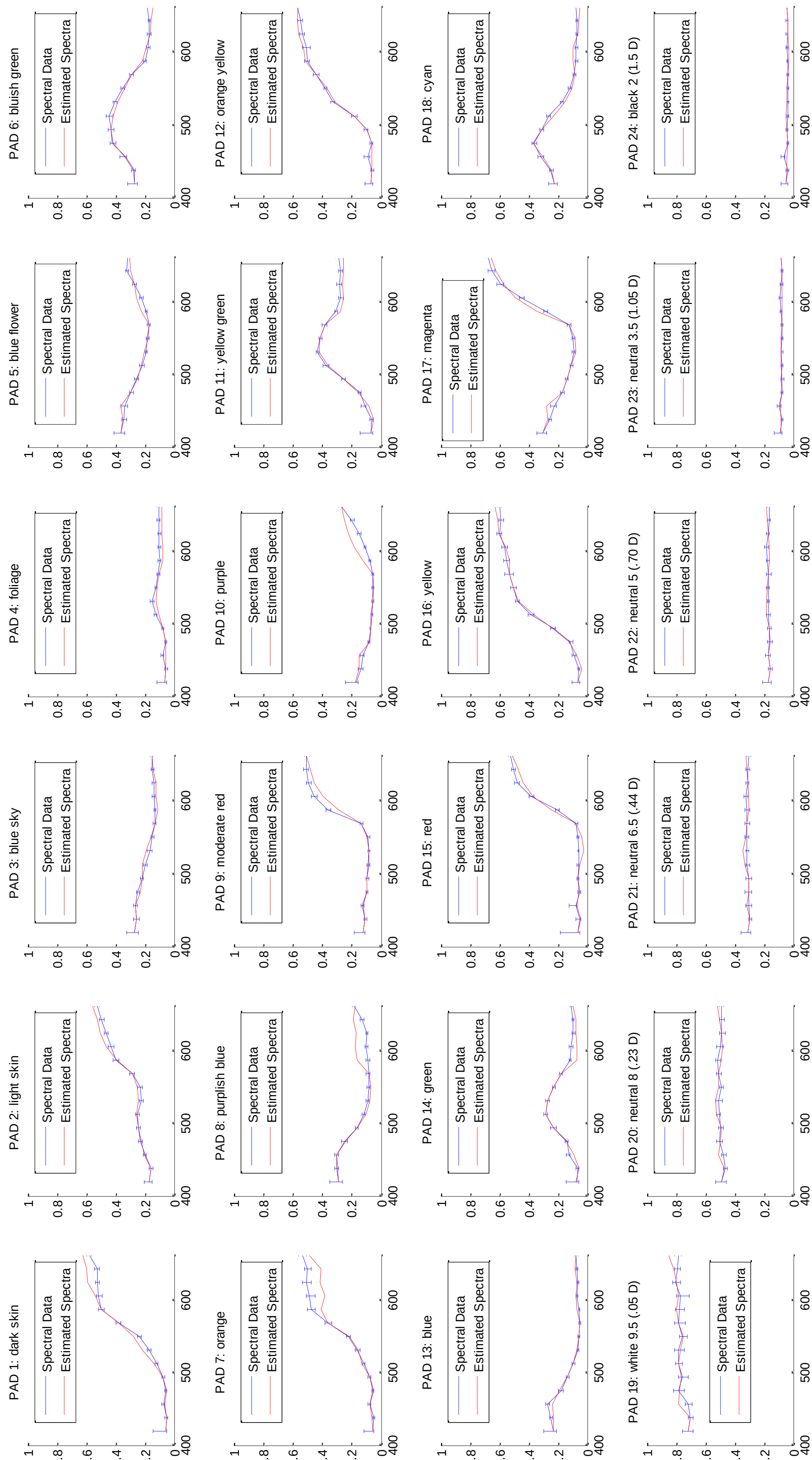


Image 21 : FFT Using NB2 & RGB



	FFT Estimation NB 1 & RGB			Thresholds		
Pad	Euclidian Norm	GFC	Spectral Angle	One	Two	Three
1	0,118758	0,997046	0,076878	1	1	0
2	0,101256	0,997547	0,07006	1	1	0
3	0,045172	0,99859	0,053109	1	1	0
4	0,067861	0,98512	0,172727	0	0	0
5	0,094289	0,996049	0,08892	1	1	0
6	0,159825	0,990804	0,135721	1	0	0
7	0,178991	0,991844	0,127808	1	0	0
8	0,198524	0,962047	0,276388	0	0	0
9	0,122663	0,996918	0,078535	1	1	0
10	0,210087	0,953272	0,306909	0	0	0
11	0,102054	0,996351	0,085457	1	1	0
12	0,136356	0,995864	0,090977	1	1	0
13	0,088383	0,986355	0,165384	0	0	0
14	0,157122	0,972908	0,233302	0	0	0
15	0,155639	0,995472	0,095196	1	1	0
16	0,073861	0,999112	0,042137	1	1	1
17	0,173131	0,995821	0,091449	1	1	0
18	0,156165	0,980593	0,197332	0	0	0
19	0,175393	0,998478	0,055177	1	1	0
20	0,093121	0,998988	0,045001	1	1	0
21	0,072708	0,998477	0,055199	1	1	0
22	0,02446	0,999368	0,035559	1	1	1
23	0,016245	0,998746	0,050089	1	1	0
24	0,026924	0,986564	0,164108	0	0	0
	Percentage Of Pads With Successful Estimation			70,83%	62,50%	8,33%

	FFT Estimation NB 2 & RGB			Thresholds		
Pad	Euclidian Norm	GFC	Spectral Angle	One	Two	Three
1	0,127969	0,998569	0,053504	1	1	0
2	0,075855	0,999245	0,038857	1	1	1
3	0,053855	0,997347	0,072861	1	1	0
4	0,0639	0,99527	0,0973	1	1	0
5	0,06508	0,998265	0,058921	1	1	0
6	0,120461	0,995179	0,098234	1	1	0
7	0,19631	0,992606	0,121684	1	0	0
8	0,169337	0,973712	0,229802	0	0	0
9	0,101786	0,997338	0,072978	1	1	0
10	0,138391	0,979073	0,204939	0	0	0
11	0,102418	0,997037	0,077003	1	1	0
12	0,054701	0,999353	0,035974	1	1	1
13	0,046386	0,996492	0,083788	1	1	0
14	0,07863	0,993033	0,11811	1	0	0
15	0,10044	0,996011	0,089351	1	1	0
16	0,055723	0,999505	0,031463	1	1	1
17	0,105248	0,997704	0,067772	1	1	0
18	0,079831	0,994801	0,102016	1	0	0
19	0,129543	0,999319	0,0369	1	1	1
20	0,095013	0,999054	0,043496	1	1	1
21	0,060647	0,998949	0,045849	1	1	0
22	0,046218	0,99782	0,066037	1	1	0
23	0,021822	0,997837	0,065792	1	1	0
24	0,034792	0,976789	0,215875	0	0	0
Percentage Of Pads With Successful Estimation				87,50%	75,00%	20,83%

So after evaluating the above results we get the following table,

	Table 5	**Thresholds**		
No	Algorithm	One	Two	Three
1	FFT	70,83%	58,33%	8,33%
2	FFT NB 1 & RGB	70,83%	62,50%	8,33%
3	FFT NB 2 & RGB	87,50%	75,00%	20,83%

From the results of Table 5 we can see that with the use of Narrow Band Filters & RGB we got better results over RGB and NB sets. We saw that with the combination of NB1 and RGB we got approximately the same results as with RGB input. But, we had slightly better fitting in the plots than RGB. So we see again that the use of six bands (3 narrow bands and 3 broadband RGB) give better results over the use of single NBs and RGB.

	Table 6	Thresholds		
No	Algorithm	One	Two	Three
1	Wiener NB 1 & RGB	83,33%	79,17%	20,83%
1	FFT NB1 & RGB	70,83%	62,50%	8,33%
2	Wiener NB 2 & RGB	87,50%	79,17%	33,33%
2	FFT NB 2 & RGB	87,50%	75,00%	20,83%

From Table 6 above we can see that Wiener also performs better over FFT. So from the analysis we made so far we can safely conclude that the use of Wiener Estimation Method is better over FFT and can provide us with a very good estimation.

4.4 <u>Six Narrow Band Filter</u>

Another experiment we could make was to use only band pass filters and not RGB. The main difference between the band pass filters and RGB is that band pass are narrowband in contrast with RGB which isn't.

Again this time we had a 24x6 matrix which now contained six band filter values. To construct the six band filter we simply used both of the Narrow Band filter sets mentioned in the previous section. Since the data we want to estimate is spectrum, we expected that Six Band filter would give better results since all input are spectra and not RGB values. As usual we checked first the Wiener and the FFT estimation methods.

Also by using six bands (2 sets of NBs), if we had good results as before with six bands (RGB and NBs) we would be able to say that the use of six bands in general can give very good estimates for Spectra.

Moreover, since we only have six bands and the time for the estimate is at the point of ms, we can get a camera with six bands to take snapshots and then with online processing, we can create a "Real Time" unit that can provide us with a spectral cube in little time.

So, we present the results for Six Narrow Band filters for both Wiener and FFT algorithms.

Image 22 : Wiener Estimation Using Six Narrow Bands

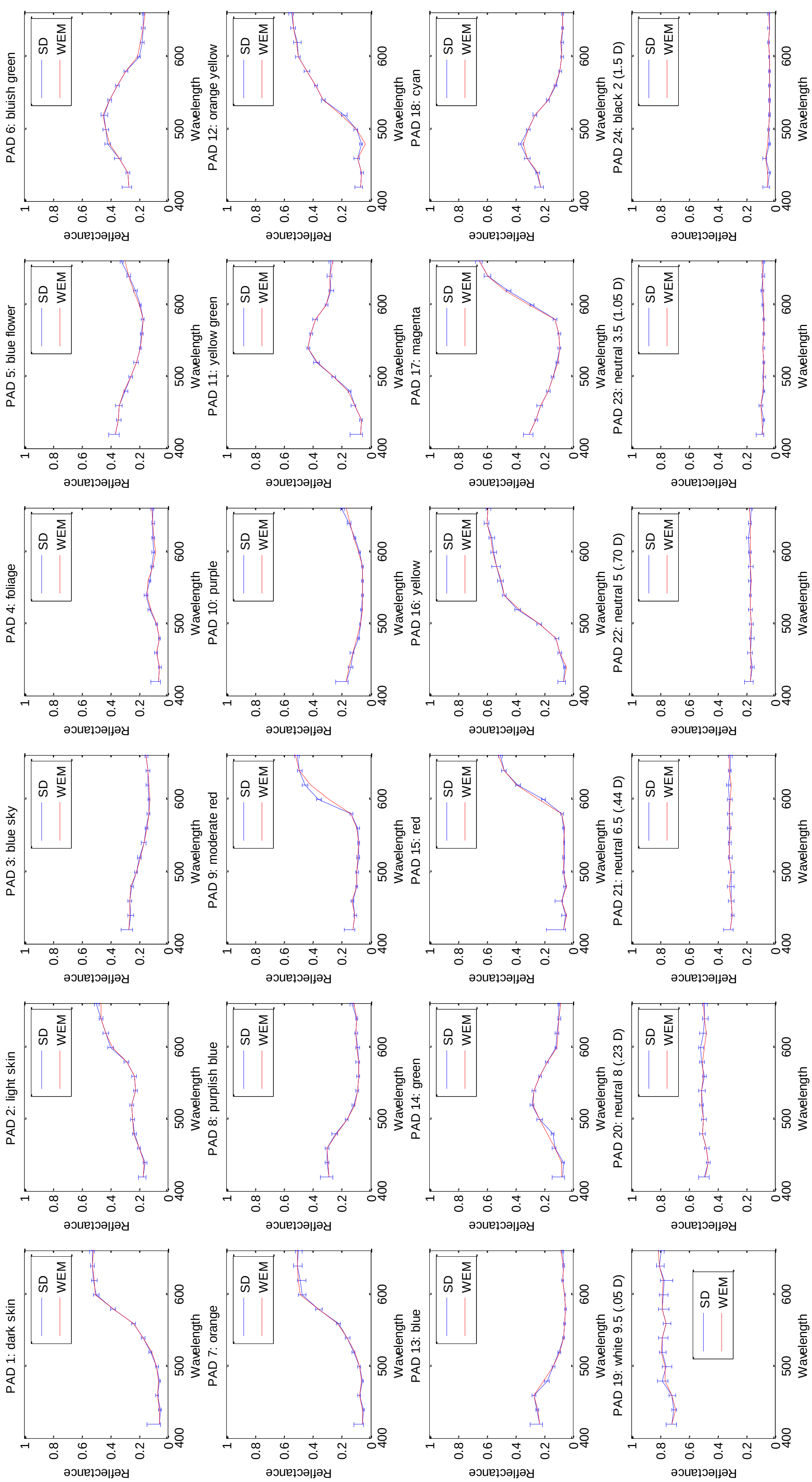
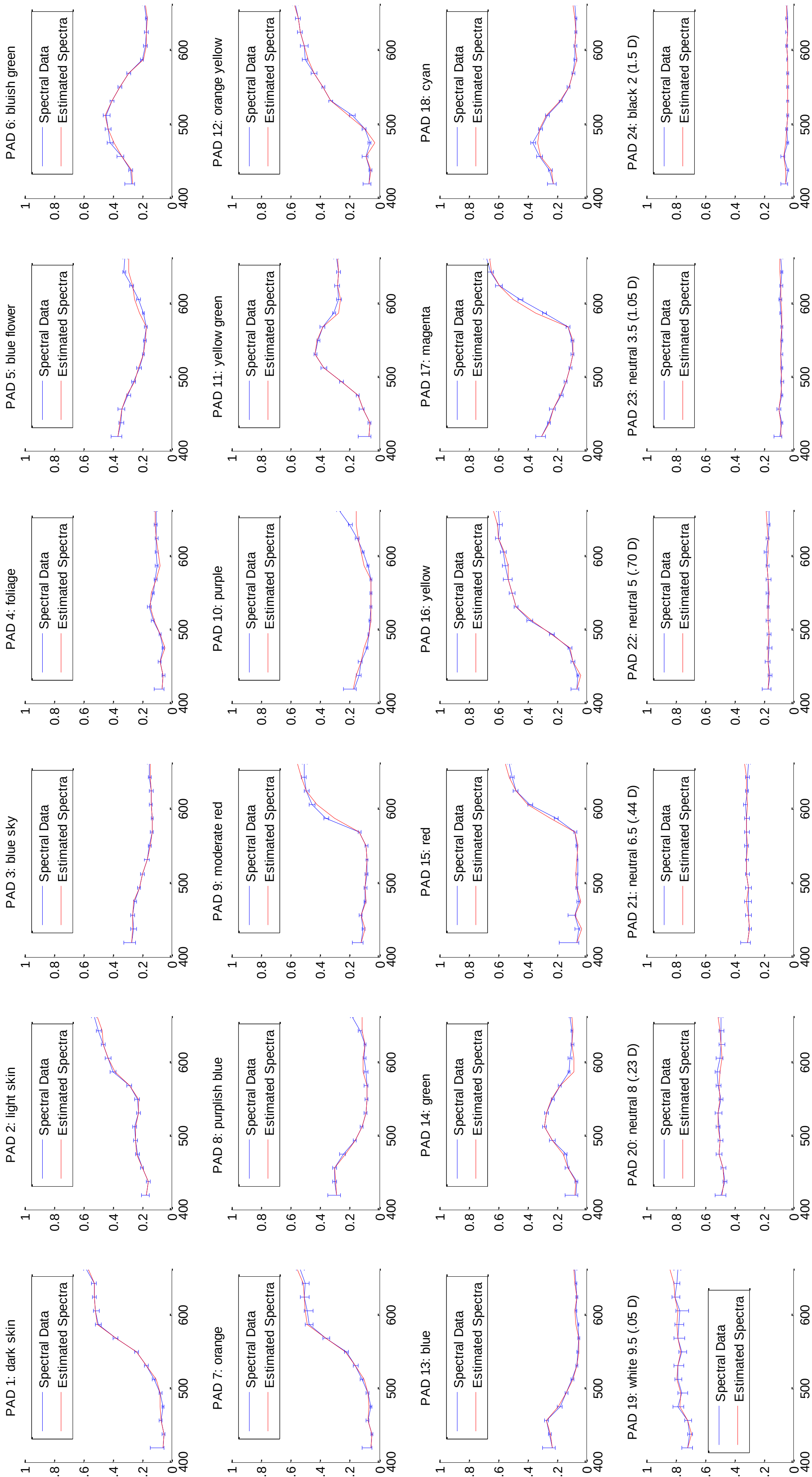


Image 23 : FFT Estimation Using Six Narrow Bands



	Wiener Estimation Six Narrow Bands			Thresholds		
Pad	Euclidian Norm	GFC	Spectral Angle	One	Two	Three
1	0,068135	0,99927	0,038206	1	1	1
2	0,105306	0,998147	0,06089	1	1	0
3	0,023648	0,999495	0,031796	1	1	1
4	0,027972	0,99742	0,07185	1	1	0
5	0,037544	0,999393	0,034835	1	1	1
6	0,065456	0,998518	0,05445	1	1	0
7	0,024537	0,999884	0,015238	1	1	1
8	0,116565	0,988671	0,15067	0	0	0
9	0,09889	0,996713	0,081107	1	1	0
10	0,215745	0,951066	0,314128	0	0	0
11	0,06121	0,998626	0,052436	1	1	0
12	0,048472	0,999508	0,031361	1	1	1
13	0,045314	0,996907	0,078671	1	1	0
14	0,069035	0,994175	0,107985	1	0	0
15	0,05351	0,999353	0,035987	1	1	1
16	0,022067	0,999921	0,012561	1	1	1
17	0,032476	0,999772	0,021334	1	1	1
18	0,025993	0,999431	0,033722	1	1	1
19	0,067231	0,999787	0,020636	1	1	1
20	0,078759	0,999192	0,040191	1	1	1
21	0,056277	0,999025	0,044157	1	1	1
22	0,037489	0,998693	0,051134	1	1	0
23	0,022765	0,998003	0,063205	1	1	0
24	0,023985	0,991898	0,12738	1	0	0
Percentage Of Pads With Successful Estimation				91,67%	83,33%	50,00%

	FFT Estimation Six Narrow Bands			Thresholds		
Pad	Euclidian Norm	GFC	Spectral Angle	One	Two	Three
1	0,062678	0,999257	0,038539	1	1	1
2	0,073815	0,999098	0,042468	1	1	1
3	0,017068	0,999747	0,022494	1	1	1
4	0,038849	0,995015	0,099887	1	1	0
5	0,06634	0,99806	0,062292	1	1	0
6	0,036858	0,99956	0,029681	1	1	1
7	0,038872	0,999751	0,022311	1	1	1
8	0,139724	0,983137	0,183905	0	0	0
9	0,093392	0,997239	0,074329	1	1	0
10	0,246706	0,92732	0,383608	0	0	0
11	0,033645	0,999591	0,02859	1	1	1
12	0,065981	0,999089	0,042684	1	1	1

13	0,031032	0,998444	0,0558	1	1	0
14	0,049469	0,997013	0,077305	1	1	0
15	0,078325	0,998778	0,049439	1	1	0
16	0,058224	0,999488	0,032006	1	1	1
17	0,091242	0,998187	0,060233	1	1	0
18	0,069171	0,995943	0,090105	1	1	0
19	0,088305	0,999663	0,025965	1	1	1
20	0,084711	0,999109	0,042207	1	1	1
21	0,059209	0,99898	0,045177	1	1	0
22	0,039406	0,998602	0,052886	1	1	0
23	0,023028	0,997898	0,064847	1	1	0
24	0,022393	0,992921	0,119059	1	0	0
Percentage Of Pads With Successful Estimation				91,67%	87,50%	41,67%

	Table 7	Thresholds		
No	Algorithm	One	Two	Three
1	Wiener Six Narrow Bands	91,67%	83,33%	50,00%
2	FFT Six Narrow Bands	91,67%	87,50%	41,67%

So, we see that the six narrow band filters gave better results to the estimation for both algorithms. From the results of Tables 6 we could see that the results we got were better over all other methods of input. From the plots we could see that we had a very accurate fit in most of the cases, thus a very accurate spectral estimations. We could also see the high percentages on the thresholds. From that and from the plots we could suggest that both algorithms perform about the same. But Wiener Estimation performed a little better since it gave better accuracy. Something that can be seen in the plots (ex. Pad 10). The main reason that we got very good results with the six narrow bands is that we got better spectral information a priori, since all the bands we used gave spectral information. And, because we had six bands we had information for a large part of spectrum in comparison with the simple 3 Narrow Bands set where we got very narrow part of the Spectral information.

Concluding, the best algorithm to use for Spectral Estimation and therefore later tried to calculate the spectral cube with is Wiener Estimation Method. That's because despite the fact the results were approximately the same, Wiener also gave the results needed faster and in some cases with a little more accuracy than FFT. Moreover, we saw that we can get very good spectral estimation using Six Bands in general (since it provided good results for both cases , six narrow – 3 narrow_3 broad). Also, in the next steps we used the Six Narrow Bands for the calculation of the Spectral Cube since this method gives by far the best results.

4.5 Calculation of Spectral Cube

The next step, after choosing the type of input that gives the best results, was to create the spectral cube. The main differences from the method above was our input in image(s). For this reason we used the MUSIS HS camera which gave us spectral images of the object wanted. So when we want to put simple RGB as an input we only use the RGB pictures taken. If we want to use a triple band filter or both we use the spectral image which corresponds to the wavelength needed. So the input will be one, four or six images.

The first thing we had to do was to determine the transformation of the images needed so we could input our data to the algorithms. The main problem was that we wanted two-dimensional data for the algorithms and the pictures were three-dimensional. So to solve this problem we came up with a reshaping idea that transforms the three-dimensional matrix of the picture to a two-dimensional matrix that we can use as input to the algorithms. So after doing the reshaping process (it needs to be mentioned that this procedure consumes enough time) we run the algorithms to calculate the spectral reflectance for each pixel of the image. After calculating the spectra of all the pixels of the image we needed to present those data in the form of pictures. To do that, we came up with the inverse procedure of the reshaping that we allows us to make a two-dimensional matrix into a three-dimensional matrix that we can use to produce a picture. So after re-reshaping the estimated matrix we could produce the spectral cube we want. The spectral cube images have the same integral as the input provided for the spectra. So if we have a 10 nm integral in the a priori spectra data, then the spectral cube created will also have a 10 nm integral. That means that for the area of 420-700 nm we will produce 29 pictures.

The main reason we want to calculate the spectral cube is to see if we can use the Spectral estimation process to get valid results as good as from a spectral imaging device. So in order to see if the results we have, are good we must compare the images calculated from the estimation process with the spectral images taken from the MUSIS HS camera.

So after calculating the spectral cube, we compared all the pictures created with the corresponding pictures taken from the MUSIS HS. To compare the two pictures we used two methods. The first one was to subtract the estimated picture from the captured one. If the resulting image is completely black, that means that the two pictures are identical, so the results provided are very good and we can use them (estimation method) instead of a spectral imaging device. The second one was to calculate the difference percentage by subtracting the two images and then divide the result with the original picture to see the percentage of the difference. If the percentage is 0% then the two images are identically (the resulting image of the first method is a black image), if it is >0% then the two images are not identical. In some cases this might be noticeable with the first method and others not. So we use the percentage to be sure.

After completing the calculation process of the spectral cube the next thing to do was to check the results for the four different types of input we can use. To test the accuracy of the image created we compared randomly the image of 520nm taken from MUSIS HS (Image 24) with the corresponding image from estimation. The first test we did was by using simply the RGB values.

Image 24 : MUSIS HS 520 nm

Image 25 : HyperSpectral Image – Multispectral Estimation With RGB input

From the results of the estimation with RGB (Image 25) we can see that we don't have matching images. The D% = 13.37%. So we see that we have a significant difference in the calculation of the cube.

Next check was the RGB with the narrow band filters. In this particular type of input we will try to estimate each of the 4 pads separately in order to avoid the "noise" that the black frames around the pads may create. So, for this particular case we show 4 pictures for each of the 2 case of Narrow Band and RGB combinations.

So, in the first estimation made using RGB and Narrow Bands set 1 we have that pad1(top left) has a difference from the original spectral about 7.12%, pad 2 (top right) has a difference of 15.59% , pad 7(bottom left) 11.62% and pad 8 14.59%. From the above percentages and from the images (Image 26) we can see the difference between the two spectral Images.

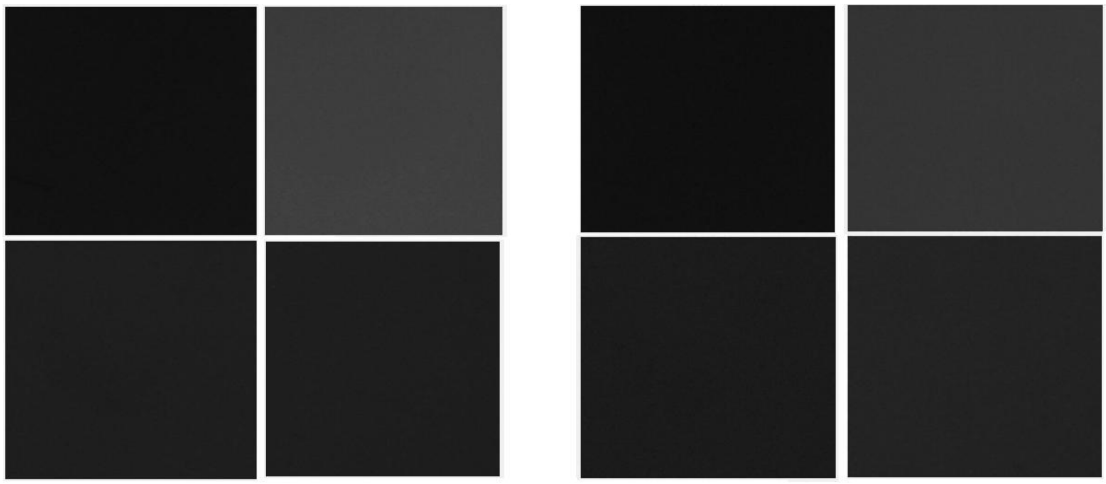


Image 26 : HyperSpectral Image – Multispectral Estimation using RGB & NB1

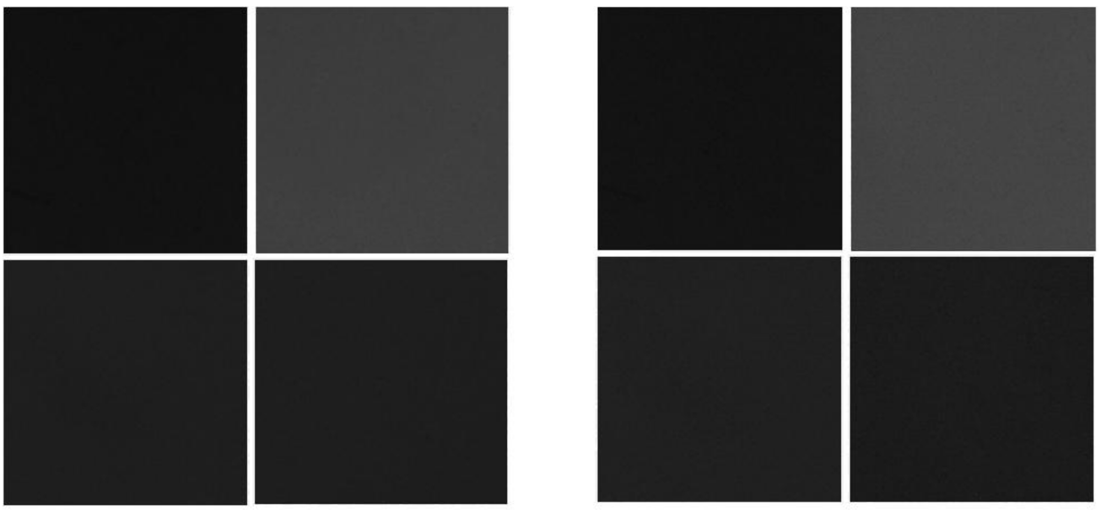


Image 27 : HyperSpectral Image – Multispectral Estimation using RGB & NB2

Next, we got the results of the combination of RGB and Narrow Band Pass Filters set 2. As before we have the measured spectra on the left and the estimated on the right. From the image (Image 28) we can see that there were color diversities. The difference percentages of the estimated over the original were for pad1 8.36%, for pad2 9.18%, for pad7 5.6% and finally for pad8 9%. As we can see all the difference percentages were below 10% which means that we get better accuracy in Spectral Cube Calculation while have input data from RGB and Narrow Band Pass Filters set 2 over the use of NB1.

Finally, the last type of input we needed to check is the six-band filter. From what we have seen in the verification process we must expect very good results. Since we now have only spectra data we won't have the problem that we saw when we used RGB and a Triple Band.

Image 28 : HyperSpectral Image – Multispectral Estimation using Six Narrow Bands

Image 29 : Compare Six-Band D% 1.47

So, as we can see from the above results the Combination of six Narrow Band filters gives us the best results so far. That's because we have a very low difference percentage (1.74%). Also, it's quite logical to say that if we make an estimate for only one pad the results will be near zero% difference (after testing the input we got 0.60% difference). Thus, we must conclude that the best method for a proper spectral cube estimation was the use of Six Bands and in our case specifically six Narrow Bands.

4.6 Minimization of the A Priori Data

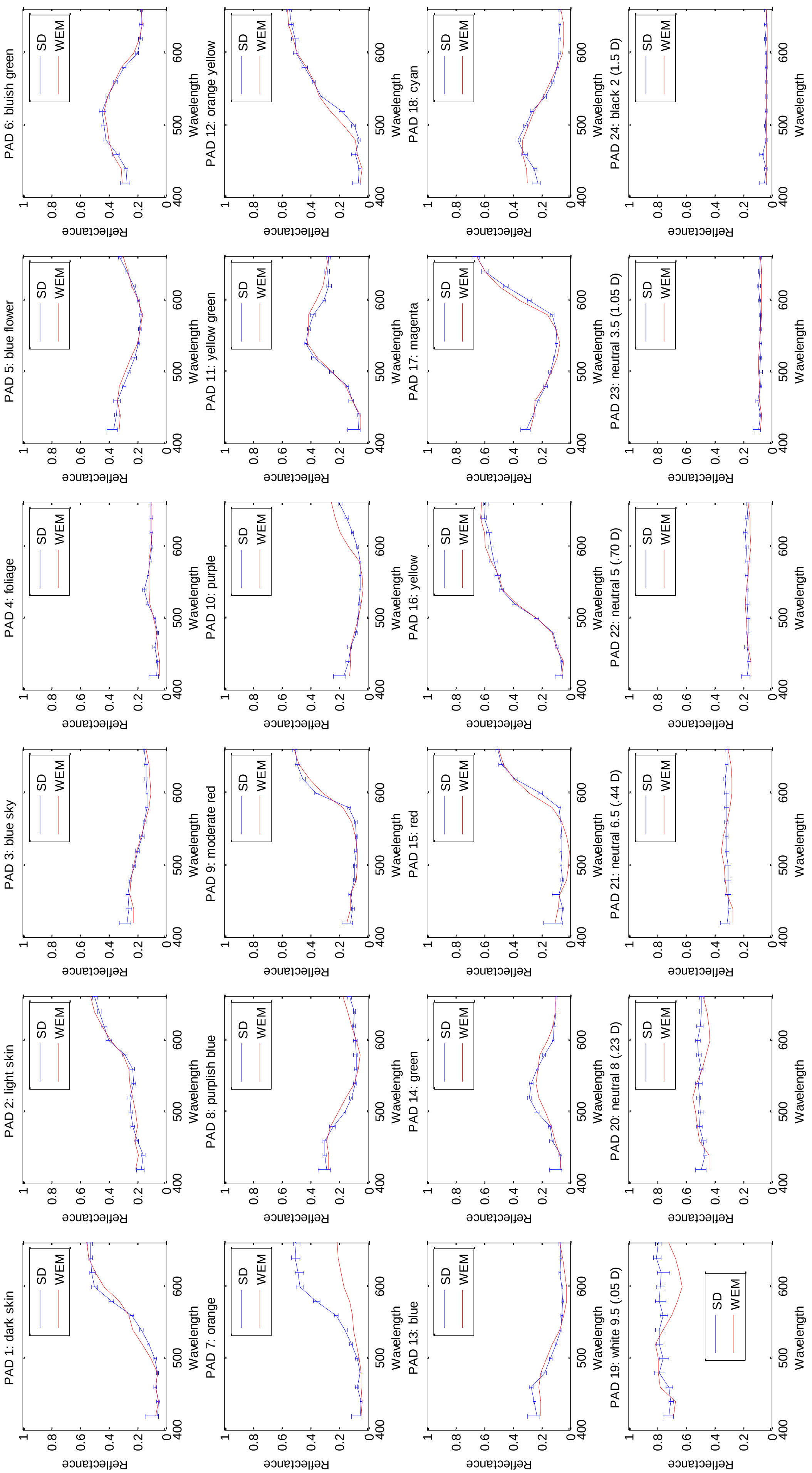
The next logical step was to see if we could minimize the input provided for the estimation. As we mentioned before, we need to have the RGB and Spectra values of each pad as a priori data. But, there is not always the possibility to have all the a priori data needed. Also, we wanted to see if we could estimate the spectra value of a (some) pad (s) without knowing nothing else but their RGB values, which means that we won't have any spectral information about them. This is called verification process. The results that we observed were the following:

- Spectral Cube
- Measures of Spectral Similarities
- Plots

So what we needed to do, in order to create this experiment was to start removing information of some pads in the MacBeth. So, first we categorized the four rows of Macbeth. The first and second row consists of colors that can be produced from the combination of primary and complementary colors. The third row contains the basic and the complementary colors and finally the fourth row contain all shades of gray, white and black. After categorizing the four rows of Macbeth we removed rows and saw the results of the estimation to conclude if we could properly estimate spectra using less a priori knowledge of the object in question.

The first row we removed was the fourth one. We ran the estimation for all kinds of input data (RGB, RGB & Triple, and Six Band) and saw the results in estimation metrics, plots and finally the Spectral Cube. Since we concluded that Wiener Estimation Method will be used for our work the following results will only be for Wiener. In addition, as mentioned above, three terms for the estimation were used.

Image 30 : Estimation Using RGB - NO 4th Row

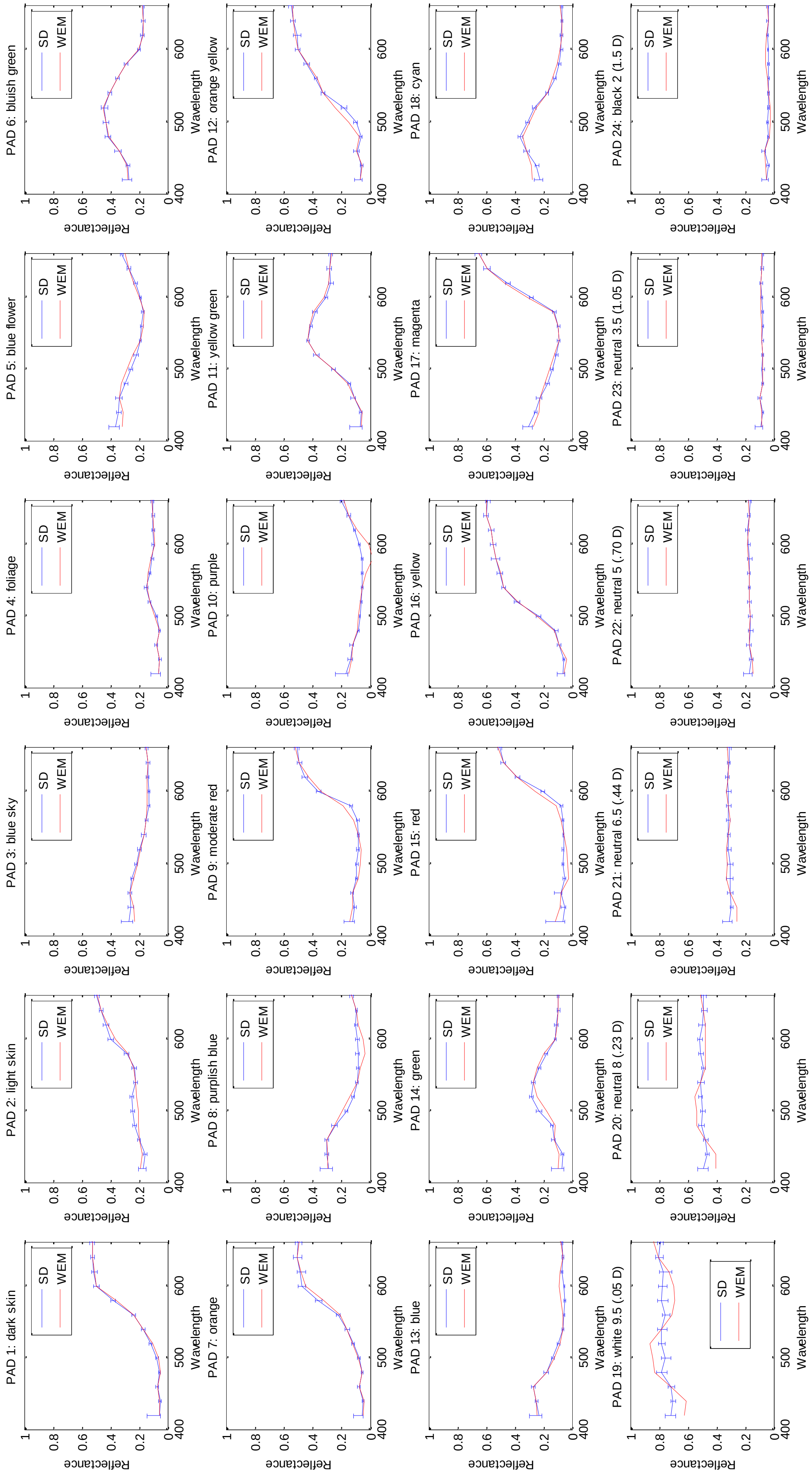


Pad	Wiener Estimation RGB (no 4th ROW)			Thresholds		
	Euclidian Norm	GFC	Spectral Angle	One	Two	Three
1	0,131615	0,9959	0,09058	1	1	0
2	0,102584	0,99763	0,06886	1	1	0
3	0,080765	0,995774	0,091964	1	1	0
4	0,051925	0,993029	0,118144	1	0	0
5	0,080073	0,997166	0,075301	1	1	0
6	0,072709	0,998272	0,058802	1	1	0
7	0,796188	0,988714	0,15038	0	0	0
8	0,112767	0,98813	0,154232	0	0	0
9	0,101817	0,99653	0,083335	1	1	0
10	0,168049	0,963863	0,269654	0	0	0
11	0,081735	0,997686	0,068035	1	1	0
12	0,128417	0,996991	0,077592	1	1	0
13	0,103581	0,982584	0,186908	0	0	0
14	0,11335	0,986809	0,162606	0	0	0
15	0,140724	0,992408	0,123301	1	0	0
16	0,098358	0,99944	0,033465	1	1	1
17	0,105969	0,998055	0,062376	1	1	0
18	0,115351	0,989588	0,144434	0	0	0
19	0,298702	0,996202	0,087178	1	1	0
20	0,180233	0,995518	0,094711	1	1	0
21	0,115684	0,995317	0,096821	1	1	0
22	0,066855	0,995111	0,098926	1	1	0
23	0,035997	0,994106	0,108626	1	0	0
24	0,03246	0,986706	0,163237	0	0	0
Percentage Of Pads With Successful Estimation				70,83%	58,33%	4,17%

		Thresholds		
No	Algorithm	One	Two	Three
1	Wiener RGB	70,83%	58,33%	8,33%
2	Wiener RGB(no 4^{th} row)	70,83%	58,33%	4,17%

As we can see from the above results, by removing the 4^{th} row we got approximately the same results using as input the RGB values but with less accuracy (Threshold 3). A thing that we must point out is that the pads removed were fairly estimated. Of course, not as good as when we had the a priori knowledge of them. Next we tried with RGB & NB 1 input.

Image 31 : Estimation Using RGB & NB1 - NO 4th Row

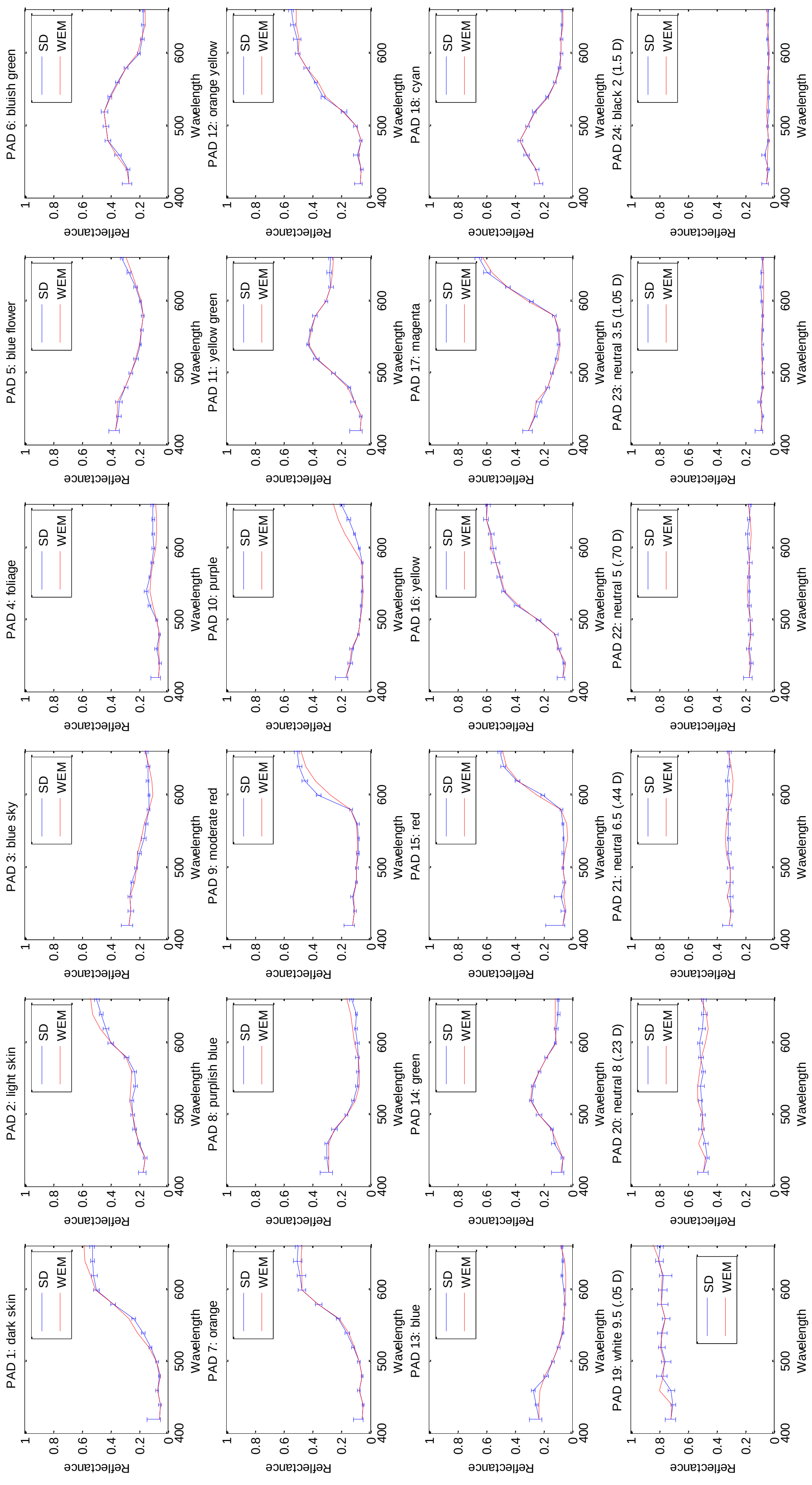


Pad	Wiener Estimation RGB & NB 1 (no 4th ROW)			Thresholds		
Pad	Euclidian Norm	GFC	Spectral Angle	One	Two	Three
1	0,078317	0,999104	0,042336	1	1	1
2	0,094762	0,998403	0,056531	1	1	0
3	0,065432	0,9961	0,088343	1	1	0
4	0,030032	0,997878	0,065153	1	1	0
5	0,081808	0,997027	0,077136	1	1	0
6	0,021516	0,999844	0,017653	1	1	1
7	0,090698	0,998911	0,046684	1	1	0
8	0,108385	0,989129	0,147587	0	0	0
9	0,101809	0,997083	0,076403	1	1	0
10	0,177608	0,971344	0,239974	0	0	0
11	0,034619	0,999546	0,030146	1	1	1
12	0,08685	0,998432	0,056001	1	1	0
13	0,065812	0,993573	0,113436	1	0	0
14	0,079568	0,992667	0,121176	1	0	0
15	0,119355	0,996055	0,088849	1	1	0
16	0,04627	0,999722	0,023589	1	1	1
17	0,070847	0,999071	0,043119	1	1	1
18	0,084265	0,994505	0,104879	1	0	0
19	0,264213	0,996107	0,088268	1	1	0
20	0,179999	0,995649	0,093323	1	1	0
21	0,10989	0,99606	0,088804	1	1	0
22	0,053601	0,997054	0,076783	1	1	0
23	0,029702	0,99729	0,073637	1	1	0
24	0,045265	0,969056	0,249421	0	0	0
	Percentage Of Pads With Successful Estimation			87,50%	75,00%	20,83%

		Thresholds		
No	Algorithm	One	Two	Three
1	Wiener RGB	70,83%	58,33%	8,33%
2	Wiener RGB & Triple 1 (no 4^{th} row)	87,50%	75,00%	20,83%
3	Wiener Triple 1 & RGB	83,33%	79,17%	20,83%

In this case of input we can see that we had a good rise of percentages on two first thresholds. That means that we got a better fit in general. So, as we can see from the thresholds table and the plots (Image 34) we got a very good fit in most pads. As it concerns the pads removed we can see that we got almost the same results on the thresholds and a little worse fitting from the plots. So, from this case we can say that a proper estimation can be made with less a priori knowledge. Next we used the same things but with NB 2 & RGB.

Image 32 : Estimation Using RGB & NB2 - NO 4th Row

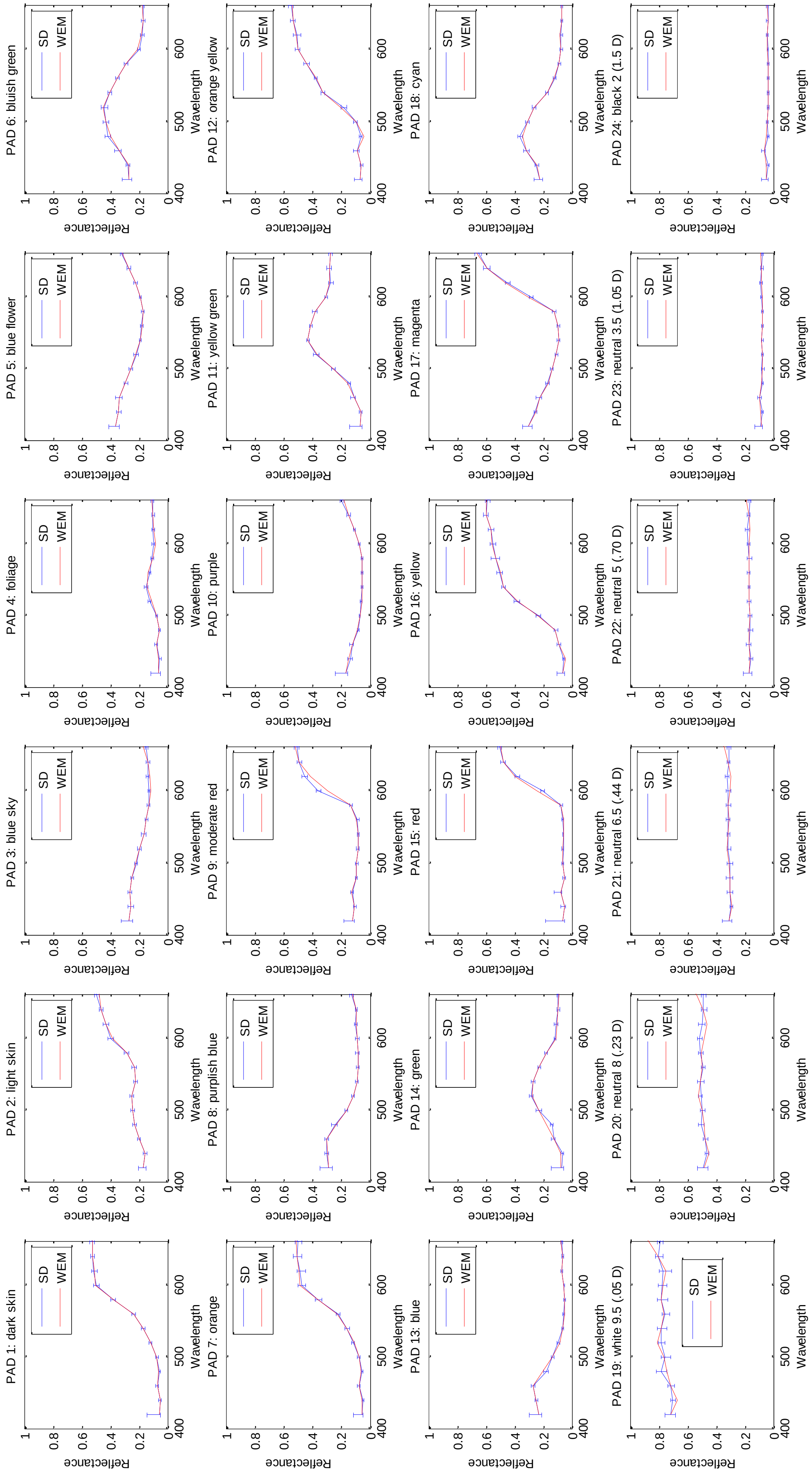


Wiener Estimation RGB & NB 2 (no 4th ROW)				Thresholds		
Pad	Euclidian Norm	GFC	Spectral Angle	One	Two	Three
1	0,10436	0,99898	0,045166	1	1	0
2	0,100245	0,998801	0,04898	1	1	0
3	0,075295	0,994942	0,10062	1	0	0
4	0,060287	0,996063	0,088768	1	1	0
5	0,048349	0,998975	0,045275	1	1	0
6	0,041118	0,999397	0,034715	1	1	1
7	0,049491	0,999775	0,021205	1	1	1
8	0,082366	0,993577	0,1134	1	0	0
9	0,12669	0,996552	0,083062	1	1	0
10	0,136534	0,977713	0,211518	0	0	0
11	0,054059	0,999144	0,041389	1	1	1
12	0,059753	0,999577	0,029084	1	1	1
13	0,063067	0,993566	0,113497	1	0	0
14	0,041234	0,998058	0,062327	1	1	0
15	0,071303	0,99819	0,060168	1	1	0
16	0,042492	0,999785	0,020727	1	1	1
17	0,058824	0,999371	0,035462	1	1	1
18	0,020654	0,999643	0,026726	1	1	1
19	0,232082	0,997882	0,065102	1	1	0
20	0,175797	0,996447	0,084328	1	1	0
21	0,10986	0,996324	0,085769	1	1	0
22	0,065826	0,995772	0,091993	1	1	0
23	0,030045	0,995661	0,093193	1	1	0
24	0,032888	0,980514	0,197733	0	0	0
Percentage Of Pads With Successful Estimation				91,67%	79,17%	29,17%

		Thresholds		
No	Algorithm	One	Two	Three
1	Wiener RGB	70,83%	58,33%	8,33%
2	Wiener RGB & NB 2 (no 4th row)	91,66%	79,16%	29,16%
3	Wiener NB 2 & RGB	83,33%	79,17%	20,83%

Here, we can see that we have also got better results in general. We can see from the plots (Image 35) that we got a very good fit even in the row we removed from the a priori data. Also, we can see that we have an 9% rise on the third thresholds which means that our estimations were more accurate. Thus, in this case by removing the fourth row we got better and more accurate results than before. The final check we did was with the six band input.

Image 33 : Estimation Using Six Narrow Bands - NO 4th Row



Wiener Estimation Six Narrow Bands (no 4th ROW)				Thresholds		
Pad	Euclidian Norm	GFC	Spectral Angle	One	Two	Three
1	0,033631	0,999821	0,018904	1	1	1
2	0,06292	0,999405	0,034507	1	1	1
3	0,08781	0,993696	0,112345	1	0	0
4	0,028718	0,997424	0,071793	1	1	0
5	0,066262	0,998463	0,055445	1	1	0
6	0,028109	0,999717	0,023802	1	1	1
7	0,033307	0,999894	0,014581	1	1	1
8	0,096328	0,992452	0,122945	1	0	0
9	0,083752	0,997613	0,069105	1	1	0
10	0,160439	0,976462	0,217398	0	0	0
11	0,026156	0,999738	0,022899	1	1	1
12	0,04161	0,999621	0,027531	1	1	1
13	0,040075	0,997666	0,06834	1	1	0
14	0,06017	0,995573	0,094133	1	1	0
15	0,05557	0,998829	0,048392	1	1	0
16	0,036891	0,999826	0,018629	1	1	1
17	0,040666	0,999835	0,018187	1	1	1
18	0,026795	0,999393	0,034853	1	1	1
19	0,314068	0,995621	0,093618	1	1	0
20	0,249013	0,992842	0,119719	1	0	0
21	0,162838	0,992517	0,122409	1	0	0
22	0,08288	0,993822	0,111216	1	0	0
23	0,029318	0,996776	0,080322	1	1	0
24	0,023718	0,991525	0,130288	1	0	0
Percentage Of Pads With Successful Estimation				95,83%	70,83%	37,50%

		Thresholds		
No	Algorithm	One	Two	Three
1	Wiener RGB	70,83%	58,33%	8,33%
2	Wiener Six Narrow Bands (no 4th row)	95,83%	70,83%	37,50%
3	Wiener Six Narrow Bands	91,67%	83,33%	50,00%

In this case we see that we got worse results in the second and third thresholds. So we don’t have as good results as with full a priori knowledge but the percentages of the thresholds are still promising. Also, from the plots (Image 36) we can see that we got an extremely good fit, which means that our estimation can be accurate without using the fourth row of MacBeth. Final step was to check the Spectral Cube production.

Image 34 : HyperSpectral Image – Multispectral Estimation With RGB input D% 14.99

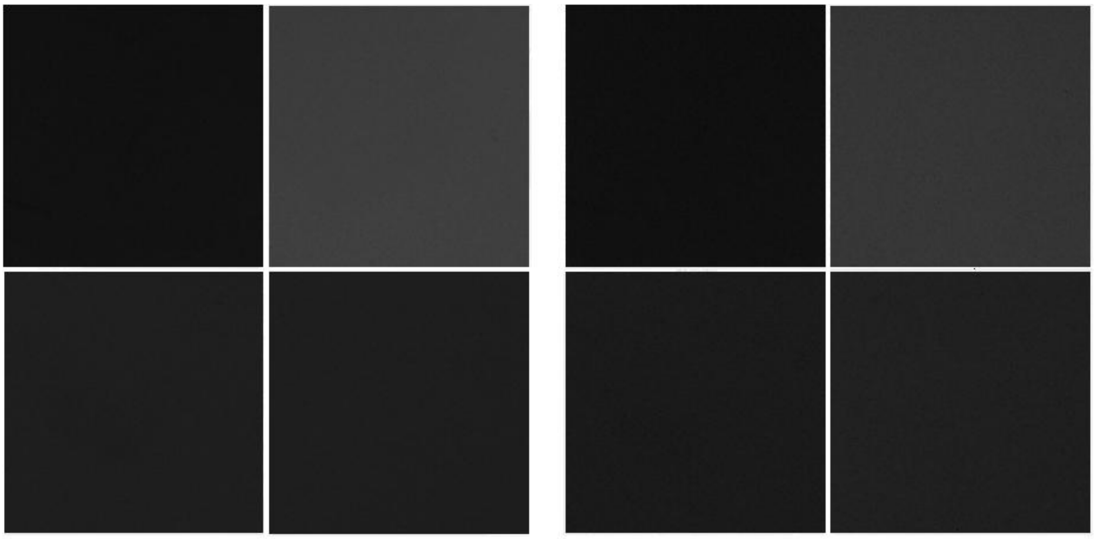


Image 35 : HyperSpectral Image – Multispectral Estimation using RGB & NB1

(no 4th row)

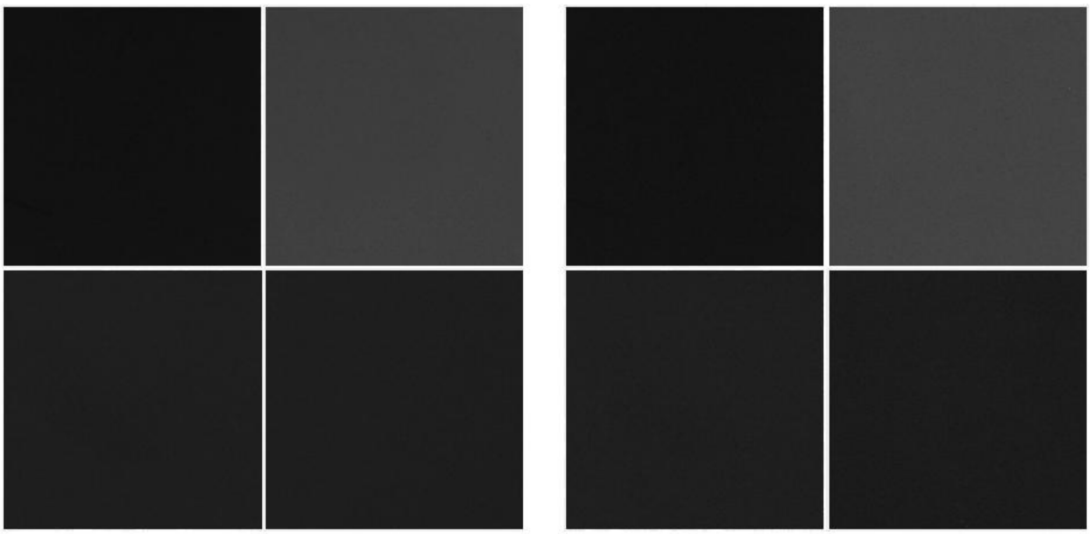


Image 36 : HyperSpectral Image – Multispectral Estimation using RGB & NB2

(no 4th row)

Image 37 : HyperSpectral Image – Multispectral Estimation using Six Narrow Bands

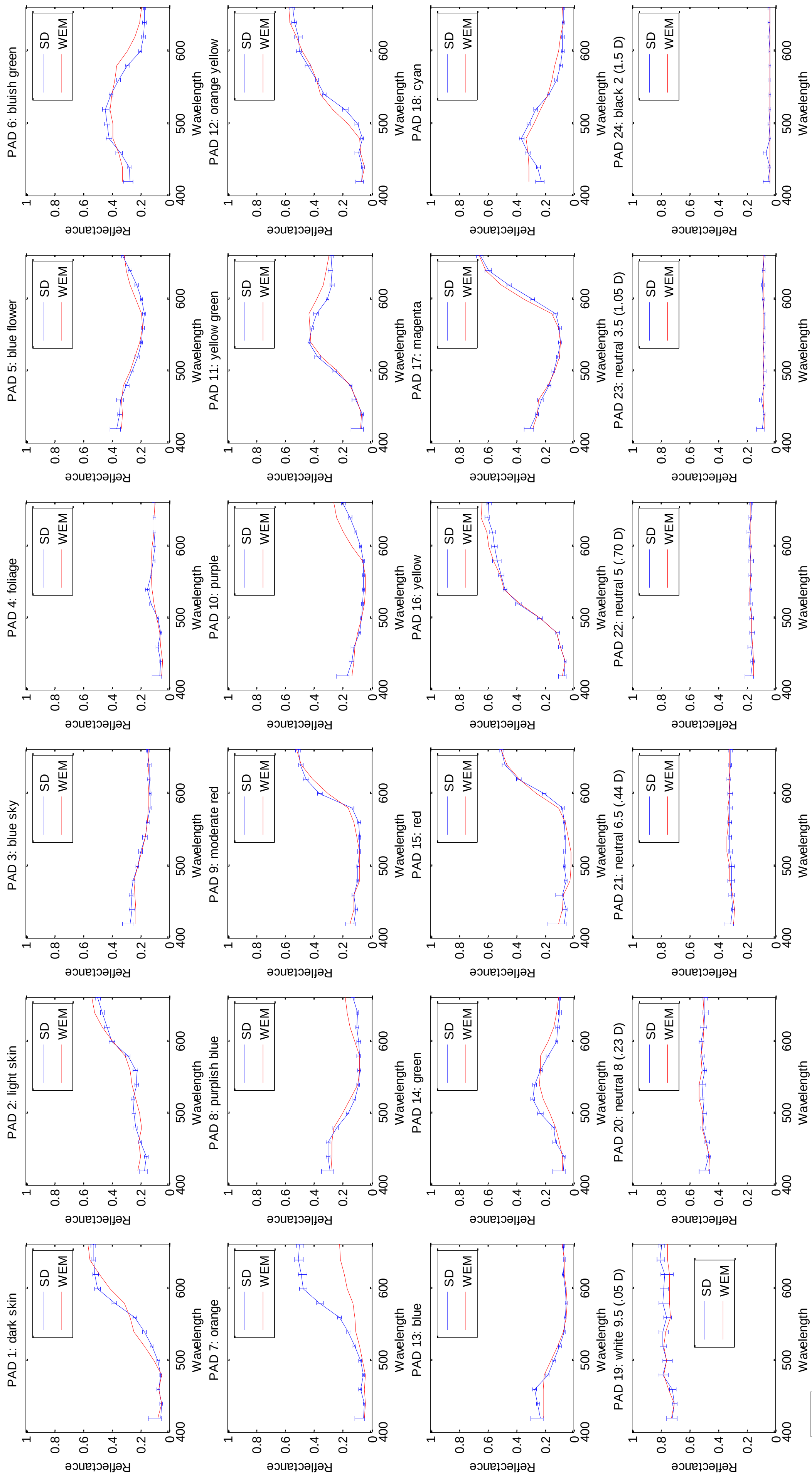
(no 4th row) D% 1.26

Observing the above results of the Spectral Cubes we can make the following remarks. The Difference percentages have fallen in two of the previous cases. In RGB & NB 2 and Six Narrow Band we had a significant fall of the percentages thus better Spectral Cube. On the other hand the RGB estimation didn't improve and RGB & NB 1 has a slight increase on some pads. We get a slightly worse image (about 1%) which is a major change. For RGB & NB 1 we had for pad1 7.37%, for pad2 17.07%, for pad7 15.31% and finally for pad8 6.23%. This difference can also be seen on Image 38 where we compared the two Spectral Images. For RGB & NB 2 we had for pad1 6.69%, for pad2 7.62%, for pad7 1.92% and finally for pad8 8.35%. and finally with the use of Six Narrow Bands Pass Filters we got 1.26% difference which means that we got a very good results as it can be seen in the comparison above (Image 40).

To conclude, the use of less a priori knowledge in this case makes our results better in some cases. So we could safely assume that we can make Spectral Estimation of Macbeth pads without using the 4th row of Macbeth.

The next step was to remove another row of Macbeth. We removed the first row leaving second and third for later. As before, we used Wiener Estimation Method with all types of input.

Image 38 : Estimation Using RGB - NO 1st Row

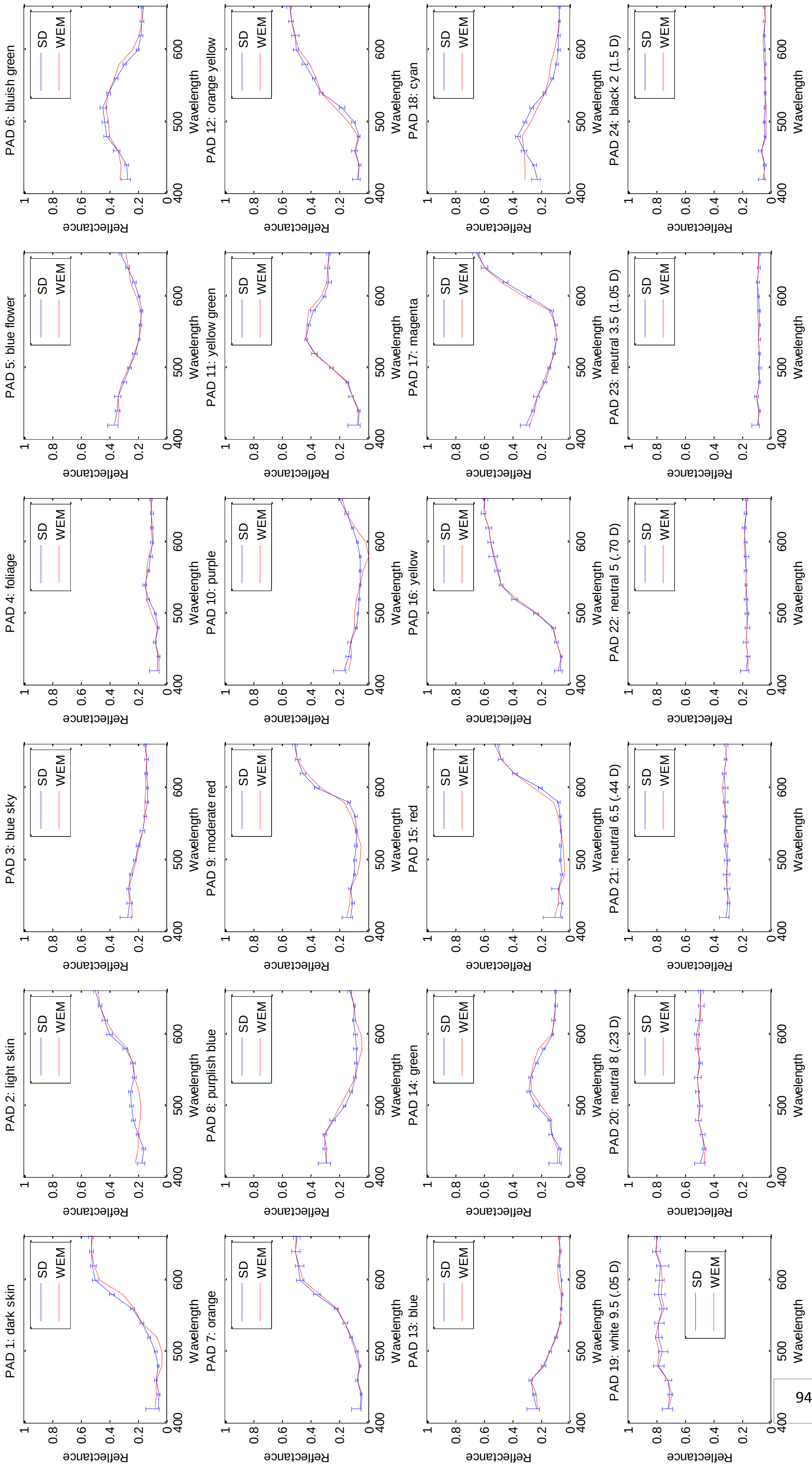


Wiener Estimation RGB (no 1st ROW)				Thresholds		
Pad	Euclidian Norm	GFC	Spectral Angle	One	Two	Three
1	0,161214	0,99376	0,111773	1	0	0
2	0,134037	0,996283	0,086251	1	1	0
3	0,057056	0,997721	0,067529	1	1	0
4	0,053659	0,991182	0,132901	1	0	0
5	0,09035	0,996892	0,078864	1	1	0
6	0,169794	0,991089	0,133601	1	0	0
7	0,793163	0,987479	0,158409	0	0	0
8	0,130221	0,984838	0,174357	0	0	0
9	0,108823	0,996067	0,088714	1	1	0
10	0,176626	0,96099	0,280237	0	0	0
11	0,124058	0,995093	0,099108	1	1	0
12	0,130018	0,997099	0,076187	1	1	0
13	0,083933	0,989027	0,148276	0	0	0
14	0,144326	0,97475	0,225197	0	0	0
15	0,110196	0,995298	0,097008	1	1	0
16	0,107428	0,999525	0,030812	1	1	1
17	0,099951	0,998578	0,05333	1	1	0
18	0,137807	0,985499	0,170507	0	0	0
19	0,112277	0,999616	0,027713	1	1	1
20	0,085557	0,999248	0,038775	1	1	1
21	0,063297	0,998932	0,046222	1	1	0
22	0,032907	0,998728	0,050436	1	1	0
23	0,02452	0,997068	0,076596	1	1	0
24	0,029605	0,987522	0,15814	0	0	0
Percentage Of Pads With Successful Estimation				70,83%	58,33%	12,50%

		Thresholds		
No	Algorithm	One	Two	Three
1	Wiener RGB	70,83%	58,33%	8,33%
2	Wiener RGB (no 1st row)	70,83%	58,33%	12,50%

As we can see from the above table we had better results without the first row. We saw that we have the same percentages on the first two thresholds and a rise on the third threshold. That means that we got better accuracy. If we look at the plots (Image 41) we will see that the fitting is almost the same with extremely small changes. So, as for RGB input the first row provided with a little better results without the use of the first row of the MacBeth. Next we checked the results using RGB & Triple Band 1 filters.

Image 39 : Estimation Using RGB & NB1 - NO 1st Row

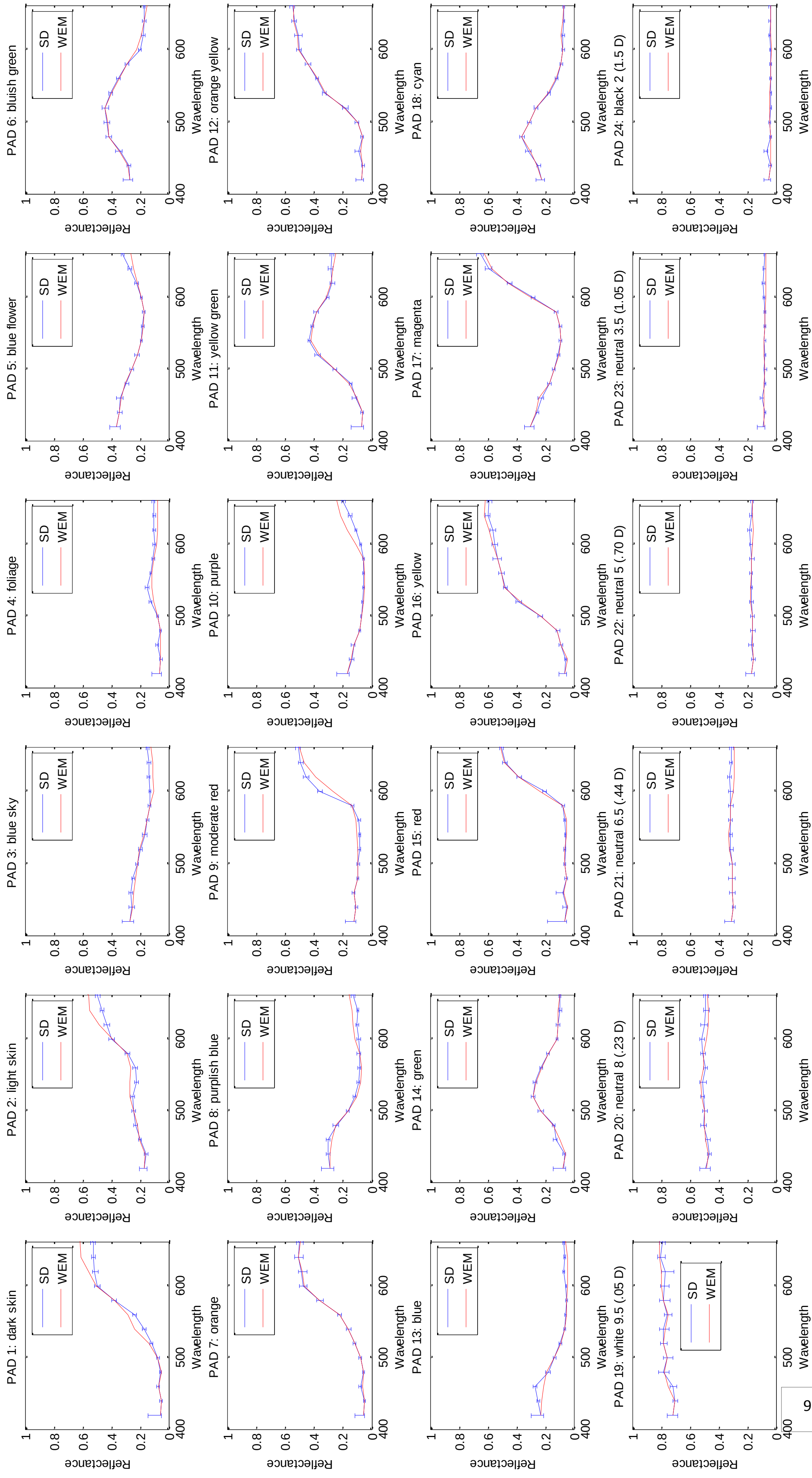


Wiener Estimation RGB & Triple 1(no 1st ROW)				Thresholds		
Pad	Euclidian Norm	GFC	Spectral Angle	One	Two	Three
1	0,236097	0,991241	0,132451	1	0	0
2	0,19145	0,992927	0,119005	1	0	0
3	0,055634	0,997261	0,074036	1	1	0
4	0,075223	0,989248	0,146772	0	0	0
5	0,065974	0,998127	0,061215	1	1	0
6	0,11507	0,995366	0,096304	1	1	0
7	0,073623	0,999276	0,038068	1	1	1
8	0,109806	0,988969	0,148669	0	0	0
9	0,088766	0,997309	0,073379	1	1	0
10	0,145307	0,980719	0,196691	0	0	0
11	0,050351	0,999079	0,042917	1	1	1
12	0,078142	0,998805	0,048889	1	1	0
13	0,055848	0,994788	0,102146	1	0	0
14	0,085821	0,99147	0,13071	1	0	0
15	0,123798	0,996589	0,082622	1	1	0
16	0,035139	0,999814	0,019291	1	1	1
17	0,057881	0,999304	0,037303	1	1	1
18	0,136834	0,98505	0,173135	0	0	0
19	0,049149	0,999866	0,0164	1	1	1
20	0,037056	0,999825	0,018726	1	1	1
21	0,03894	0,999481	0,03222	1	1	1
22	0,025951	0,999235	0,039118	1	1	1
23	0,021613	0,99842	0,05623	1	1	0
24	0,030341	0,984676	0,175292	0	0	0
Percentage Of Pads With Successful Estimation				79,17%	62,50%	33,33%

		Thresholds		
No	Algorithm	One	Two	Three
1	Wiener RGB	70,83%	58,33%	8,33%
2	Wiener RGB & Triple 1(no 1st row)	79,17%	62,50%	33,33%
3	Wiener RGB & Triple 1	83,33%	79,17%	20,83%

From the above table we can see that we a little lower results with the removal of the first row on the a priori data. We can observe that we got better results with the third threshold which means that we didnt have better accuracy on some pads. But on general we can see that the first two thresholds fell which means that we didn't have good fitting in general. Something that we can also see from the plots (Image 42). For example, in comparison with full a priori Pad 10 makes a deep around 580 nm. Next, we checked the RGB & NB 2 input.

Image 40 : Estimation Using RGB & NB2 - NO 1st Row

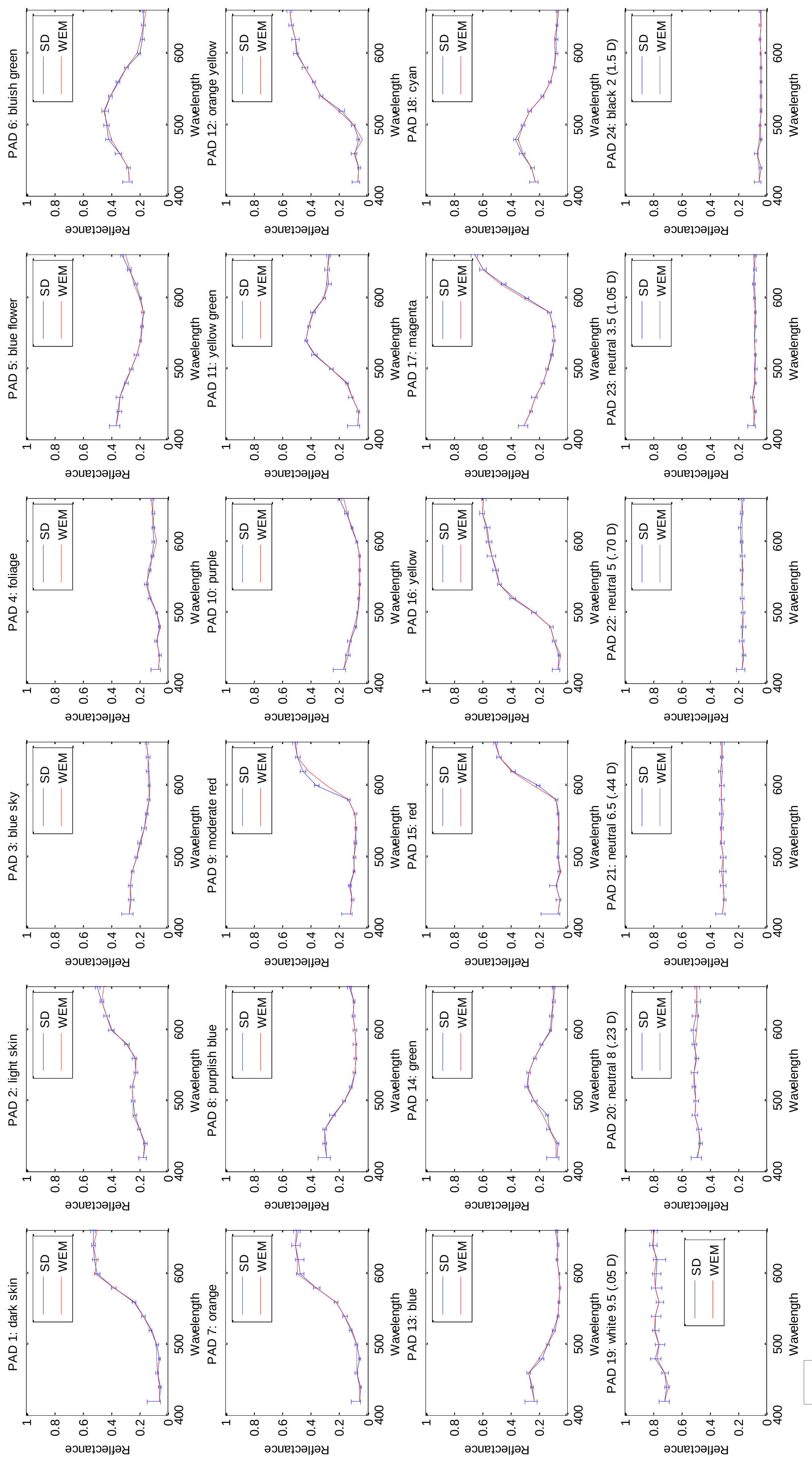


Pad	Wiener Estimation RGB & NB 2(no 1st ROW)			Thresholds		
Pad	Euclidian Norm	GFC	Spectral Angle	One	Two	Three
1	0,164424	0,997515	0,070509	1	1	0
2	0,142564	0,997271	0,073892	1	1	0
3	0,065522	0,996989	0,077627	1	1	0
4	0,079181	0,992709	0,120831	1	0	0
5	0,079572	0,997909	0,064684	1	1	0
6	0,111448	0,995632	0,093497	1	1	0
7	0,04231	0,999775	0,021228	1	1	1
8	0,114615	0,987975	0,155236	0	0	0
9	0,11677	0,995959	0,089931	1	1	0
10	0,129168	0,977989	0,210202	0	0	0
11	0,094697	0,997153	0,075474	1	1	0
12	0,03072	0,999814	0,019295	1	1	1
13	0,074519	0,993151	0,117107	1	0	0
14	0,04377	0,997881	0,065109	1	1	0
15	0,050114	0,999273	0,038141	1	1	1
16	0,057715	0,999722	0,023588	1	1	1
17	0,050939	0,999539	0,030352	1	1	1
18	0,045322	0,998258	0,059033	1	1	0
19	0,071484	0,999804	0,01978	1	1	1
20	0,072613	0,999274	0,038111	1	1	1
21	0,056555	0,998908	0,046736	1	1	0
22	0,035375	0,998525	0,054316	1	1	0
23	0,032864	0,996188	0,087346	1	1	0
24	0,164424	0,997515	0,070509	1	1	0
	Percentage Of Pads With Successful Estimation			87,50%	79,17%	29,17%

		Thresholds		
No	Algorithm	One	Two	Three
1	Wiener RGB	70,83%	58,33%	8,33%
2	Wiener RGB & NB 2(no 1st row)	87,50%	79,17%	29,17%
3	Wiener RGB & NB 2	87,50%	79,17%	33,33%

With the use of this input we can see that we had approximately the same estimation. The only difference is that we lost a small percentage of Pads from the third thresholds which means that the estimation is not so accurate as before. From the plots (Image43) we can see that we had the same fitting with the only difference that we could have a very good fit in different pads in comparison with the full data case (ex. Pad 7 gives better fitting in this case). Next check was the Six Band filter input.

Image 41 : Estimation Using Six Narrow Bands - NO 1st Row



	Wiener Estimation Six Narrow Bands (no 1st ROW)			Thresholds		
Pad	Euclidian Norm	GFC	Spectral Angle	One	Two	Three
1	0,159809	0,995782	0,091884	1	1	0
2	0,193923	0,992975	0,118605	1	0	0
3	0,032273	0,999087	0,042727	1	1	1
4	0,057423	0,990789	0,135831	1	0	0
5	0,035351	0,99945	0,033166	1	1	1
6	0,091851	0,997065	0,076632	1	1	0
7	0,088068	0,998235	0,059422	1	1	0
8	0,092265	0,99316	0,117025	1	0	0
9	0,085222	0,997558	0,069901	1	1	0
10	0,185238	0,966212	0,260691	0	0	0
11	0,046777	0,999182	0,040458	1	1	1
12	0,057729	0,999364	0,035672	1	1	1
13	0,061441	0,994377	0,1061	1	0	0
14	0,064702	0,994896	0,101075	1	0	0
15	0,040016	0,999521	0,030951	1	1	1
16	0,029845	0,999869	0,016177	1	1	1
17	0,030015	0,999832	0,018323	1	1	1
18	0,048646	0,998058	0,062336	1	1	0
19	0,034585	0,999937	0,011211	1	1	1
20	0,067034	0,999403	0,034567	1	1	1
21	0,048424	0,999264	0,038377	1	1	1
22	0,033955	0,998913	0,046639	1	1	0
23	0,027313	0,997195	0,074916	1	1	0
24	0,023327	0,992136	0,125491	1	0	0
	Percentage Of Pads With Successful Estimation			95,83%	70,83%	41,67%

		Thresholds		
No	Algorithm	One	Two	Three
1	Wiener RGB	70,83%	58,33%	8,33%
2	Wiener Six Narrow Bands (no 1st row)	95,83%	70,83%	41,67%
3	Wiener Six Narrow Bands	91,67%	83,33%	50,00%

As we can see in the final step of checking, we got better results on the first threshold but worse results in the other two. So, in a few words the estimation with the use of the first row of Macbeth gave better results. Also, the percentages were quite high, which means that we could still use this method to perform spectral estimation. In comparison with all the previous methods, Six bands provided a very accurate result. From the plots (Image 44) we can verify the accuracy of the estimated spectra. Next, we present the results of the spectral cubes for each case.

Image 42 : HyperSpectral Image – Multispectral Estimation With RGB input (No 1st row) D% 14.13

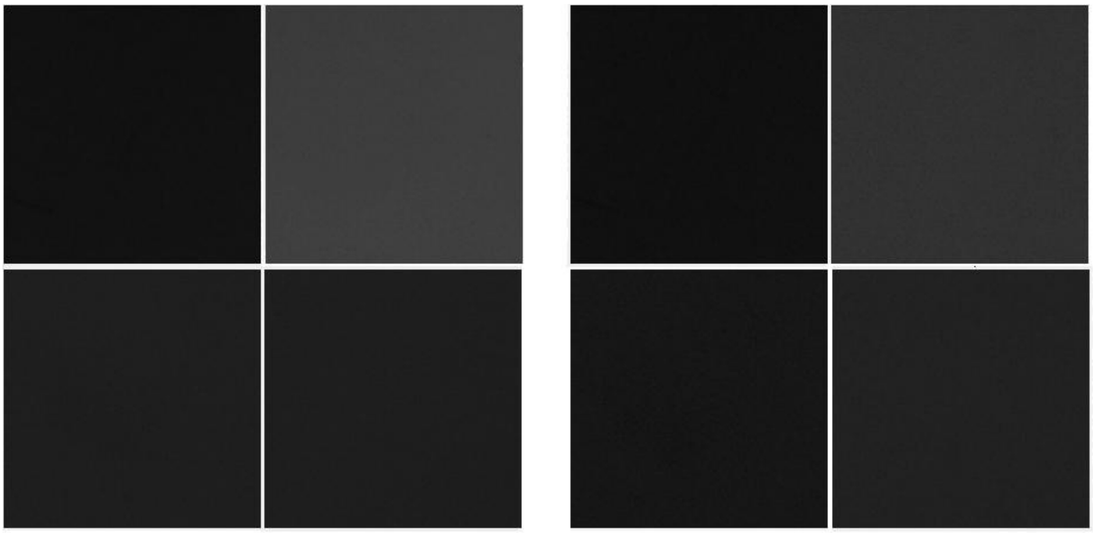


Image 43 : HyperSpectral Image – Multispectral Estimation With RGB & NB 1 (No 1st row)

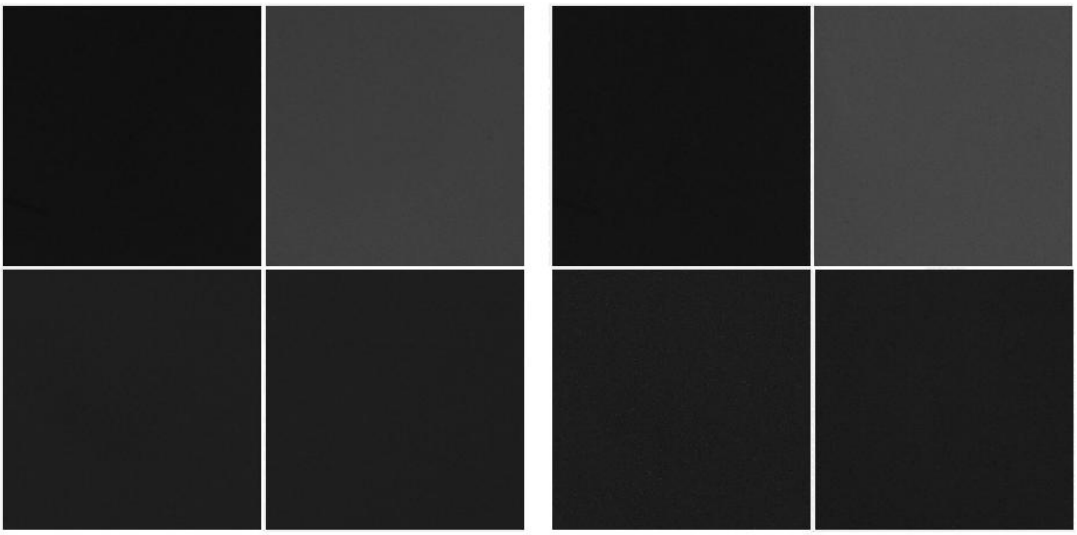


Image 44: HyperSpectral Image – Multispectral Estimation With RGB & NB 2 (No 1st row)

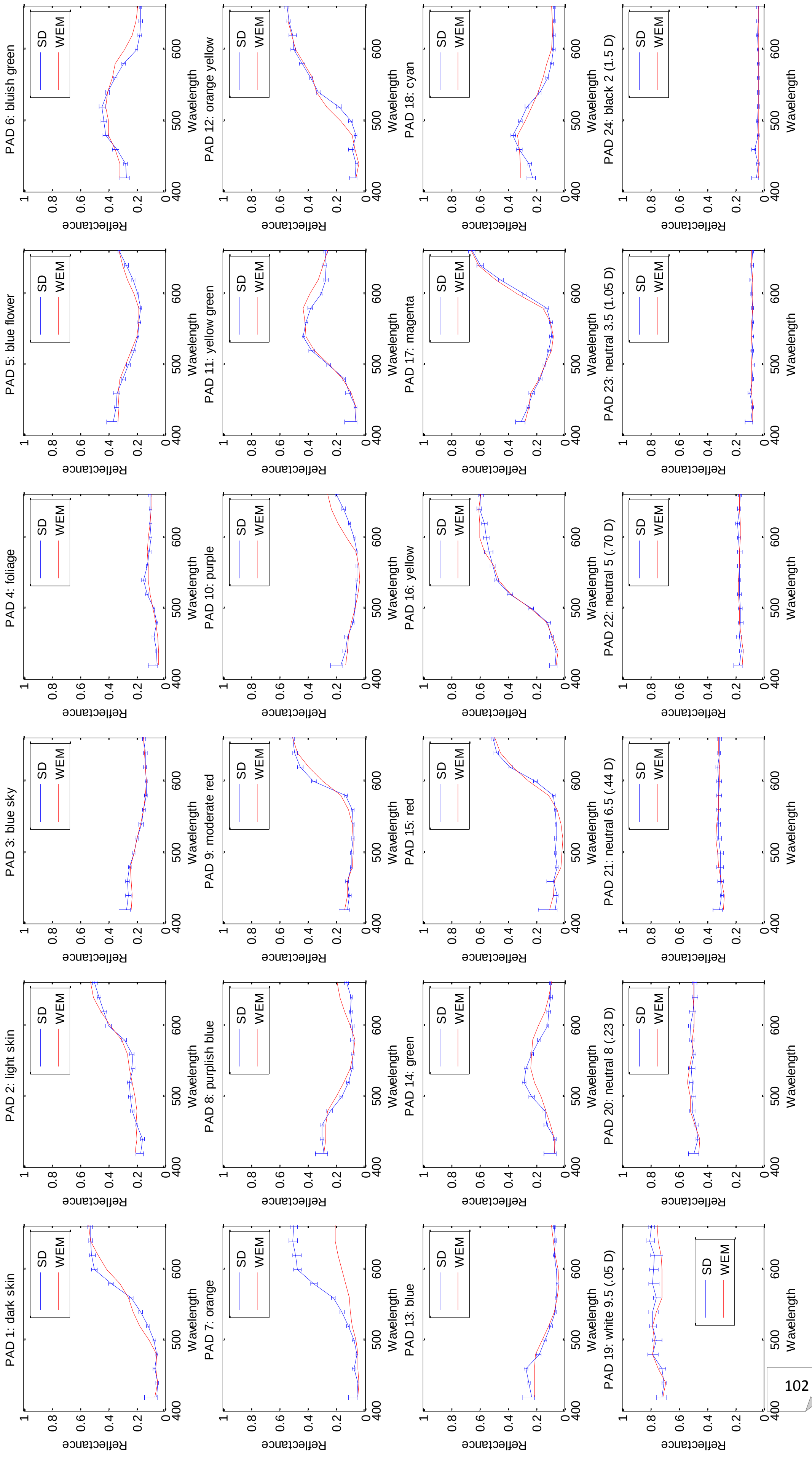
Image 45: HyperSpectral Image – Multispectral Estimation With Six Narrow Band (No 1st row) D% 1.77

For RGB & NB 1 (Image 46) we had for pad1 6.87 %, for pad2 26.23%, for pad7 35.8% and finally for pad8 10.18%. So as we can see we can get even better results than the ones from full data knowledge. We saw that only the pads on the left have a big raise on their difference. For RGB & NB 2 we had for pad1 9.11 %, for pad2 10.73%, for pad7 5.30% and finally for pad8 9.02%. We also saw a small rise and in the percentage of the NB2 set. That means that with the removal of this row we couldn’ts perform very good estimation. Since the pads in question were the pads that we removed from the a priori knowledge, it’s quite logical that they wouldn’t perform that well.

Finally with the Six Narrow Bands input we have a small raise of the difference percentage (about 0.3%). So, again we had the best results with Six Narrow Bands input.

So concluding for this case we can see that the results we took were not as good as before. Thus, we can leave out this row under certain cases. In addition to the previous conclusion about the 4^{th} row, we can have less a priori knowledge about the pads but under certain cases. So the main point here is that we can estimate elements without having a priori spectral knowledge of them using the Six Narrow Bands which is current point we wanted to make with this experiment. Next, we removed the second row.

Image 46 : Estimation Using RGB - NO 2nd Row

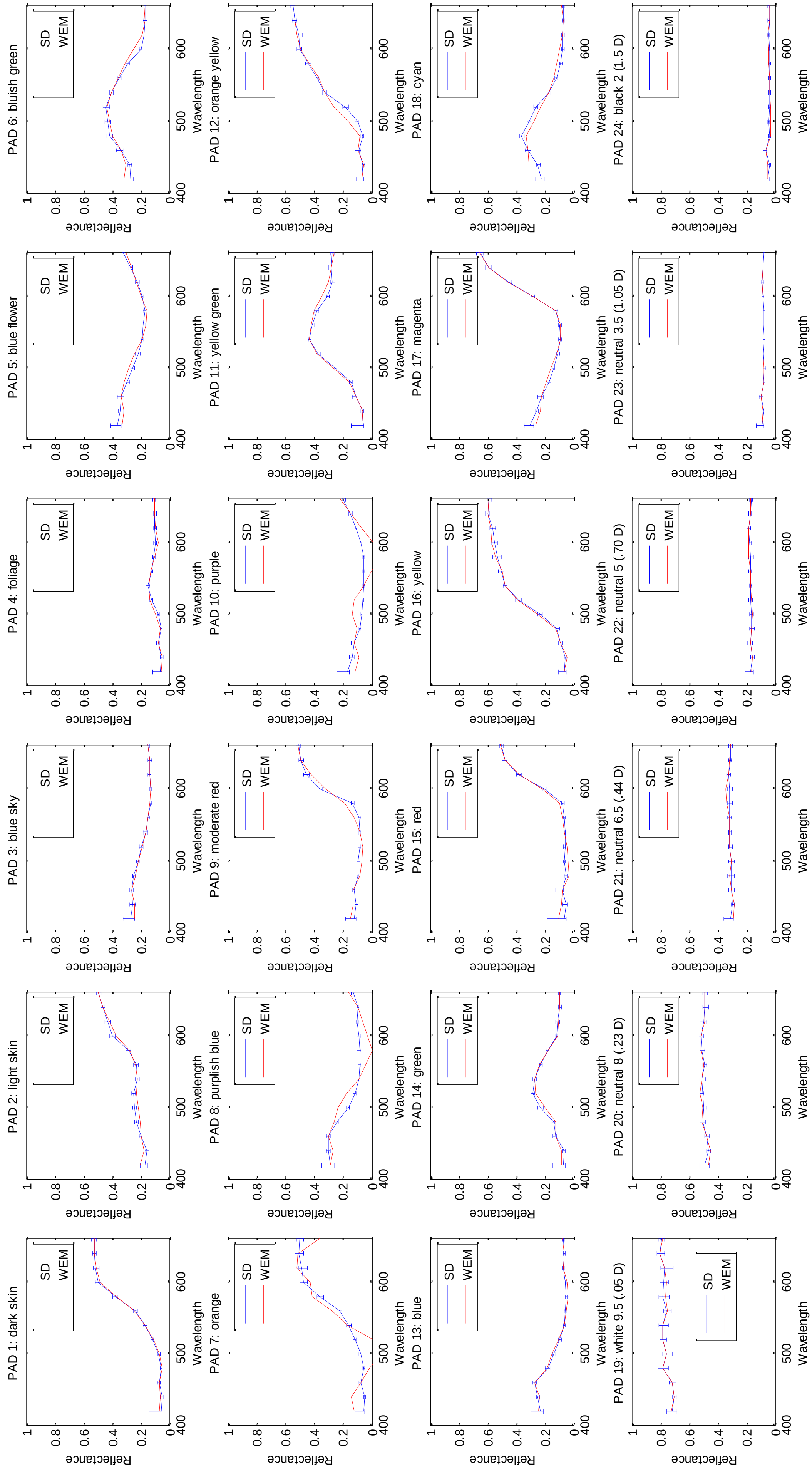


	Wiener Estimation RGB (no 2^{nd} ROW)			Thresholds		
Pad	Euclidian Norm	GFC	Spectral Angle	One	Two	Three
1	0,150577	0,995216	0,09785	1	1	0
2	0,110149	0,997102	0,076148	1	1	0
3	0,054186	0,998042	0,062588	1	1	0
4	0,057128	0,990368	0,138903	1	0	0
5	0,080509	0,997484	0,07095	1	1	0
6	0,153824	0,992802	0,120053	1	0	0
7	0,805334	0,987907	0,155673	0	0	0
8	0,143363	0,98168	0,191709	0	0	0
9	0,109059	0,99596	0,089916	1	1	0
10	0,17404	0,960915	0,280509	0	0	0
11	0,120593	0,994923	0,100812	1	0	0
12	0,130944	0,996341	0,08557	1	1	0
13	0,087241	0,987618	0,15753	0	0	0
14	0,138791	0,976944	0,215154	0	0	0
15	0,114981	0,994719	0,102821	1	0	0
16	0,076875	0,999501	0,031581	1	1	1
17	0,076652	0,998937	0,046102	1	1	0
18	0,133893	0,986619	0,163771	0	0	0
19	0,143317	0,999361	0,035762	1	1	1
20	0,088948	0,999027	0,044109	1	1	1
21	0,065729	0,998655	0,051877	1	1	0
22	0,037874	0,998367	0,05716	1	1	0
23	0,026414	0,99665	0,081874	1	1	0
24	0,030257	0,987365	0,159131	0	0	0
	Percentage Of Pads With Successful Estimation			70,83%	54,17%	12,50%

		Thresholds		
No	Algorithm	One	Two	Three
1	Wiener RGB	70,83%	58,33%	8,33%
2	Wiener RGB (no 2ndrow)	70,83%	54,17%	12,50%

Here we see that we had approximately the same estimation percentages. We have a drop on the second threshold and a small rise on the third criterion which means that we had worse estimation on a few pads and a much better accuracy on some others. Next, check was the use of RGB & NB 1 as input.

Image 47 : Estimation Using RGB & NB1 - NO 2nd Row

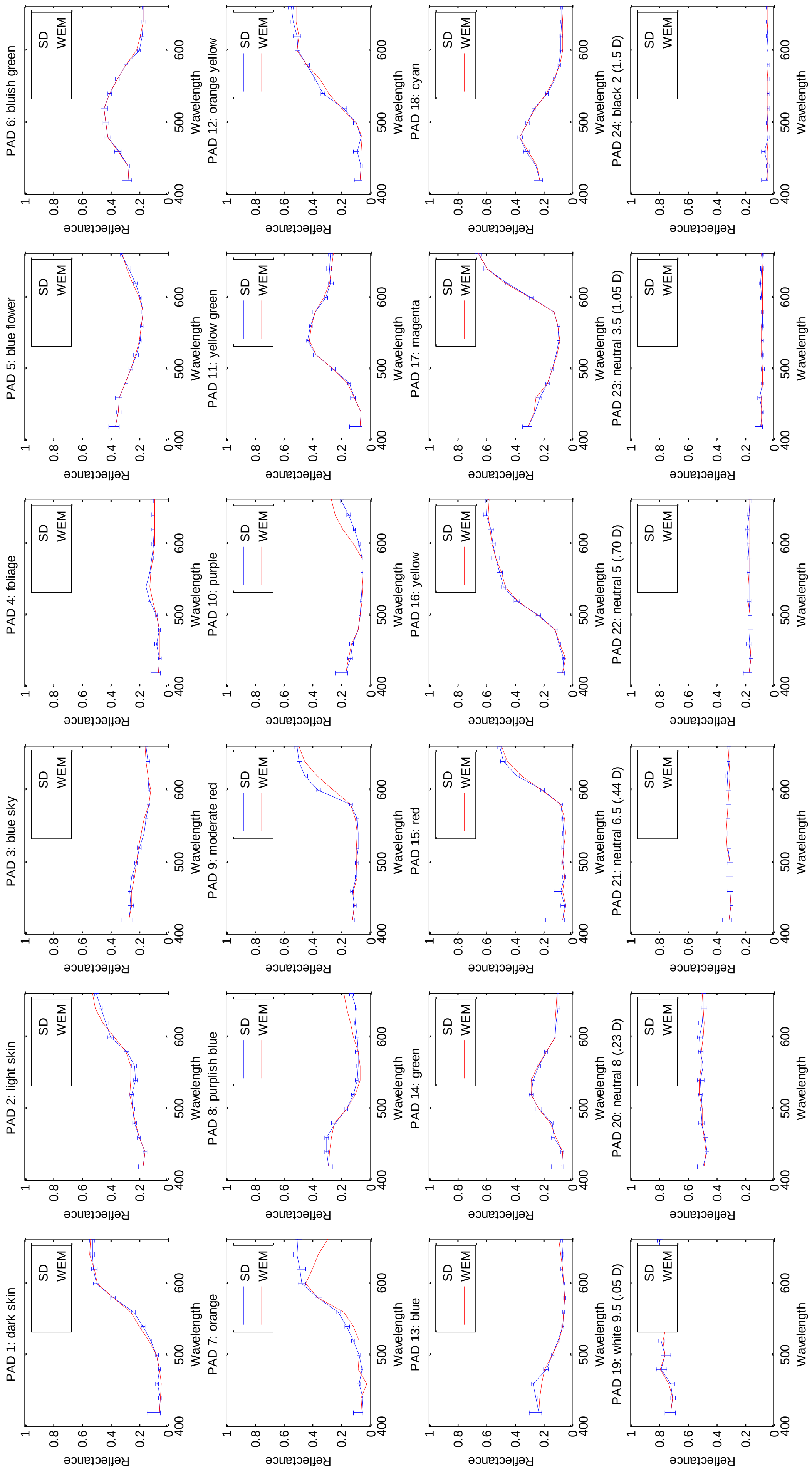


	Wiener Estimation RGB & NB 1 (no 2ndROW)			Thresholds		
Pad	Euclidian Norm	GFC	Spectral Angle	One	Two	Three
1	0,046456	0,999703	0,024386	1	1	1
2	0,08075	0,998576	0,053378	1	1	0
3	0,033007	0,999372	0,035452	1	1	1
4	0,035392	0,995869	0,090922	1	1	0
5	0,06832	0,997937	0,06425	1	1	0
6	0,094707	0,996917	0,07855	1	1	0
7	0,430748	0,951675	0,312152	0	0	0
8	0,175451	0,970502	0,243493	0	0	0
9	0,107184	0,996274	0,086347	1	1	0
10	0,223999	0,930477	0,375083	0	0	0
11	0,067826	0,998348	0,057482	1	1	0
12	0,116252	0,997133	0,075742	1	1	0
13	0,028728	0,998649	0,051996	1	1	0
14	0,059435	0,996587	0,082637	1	1	0
15	0,074958	0,997915	0,064588	1	1	0
16	0,049419	0,999801	0,019964	1	1	1
17	0,052417	0,999405	0,034495	1	1	1
18	0,12929	0,986631	0,1637	0	0	0
19	0,050209	0,999895	0,014465	1	1	1
20	0,068187	0,999419	0,034084	1	1	1
21	0,059011	0,998997	0,044781	1	1	0
22	0,021542	0,999551	0,029975	1	1	1
23	0,013248	0,999497	0,031712	1	1	1
24	0,020077	0,992374	0,123579	1	0	0
	Percentage Of Pads With Successful Estimation			83,33%	79,17%	33,33%

		Thresholds		
No	Algorithm	One	Two	Three
1	Wiener RGB	70,83%	58,33%	8,33%
2	Wiener RGB & NB 1(no 2nd row)	83,33%	79,17%	33,33%
3	Wiener RGB & NB 1	83,33%	79,17%	20,83%

From the above result we can see that we got the same results on the first two thresholds and better percentage on the third by omitting the 2nd row of MacBeth. From that assumed that we have more pads estimated accurately. From the plots (Image 50) we can see that the rest of the rows (1st,3rd ,4th) give a little better fitting thus the rise on the third thresholds. But if we compare the plots we can see that the second row doesn’t give that good estimated as it did before with full data. That means that we can get good results in general but we lose accuracy on the “unknown” samples we want to estimate. Next, RGB & NB 2 was used as input.

Image 48 : Estimation Using RGB & NB2 - NO 2nd Row

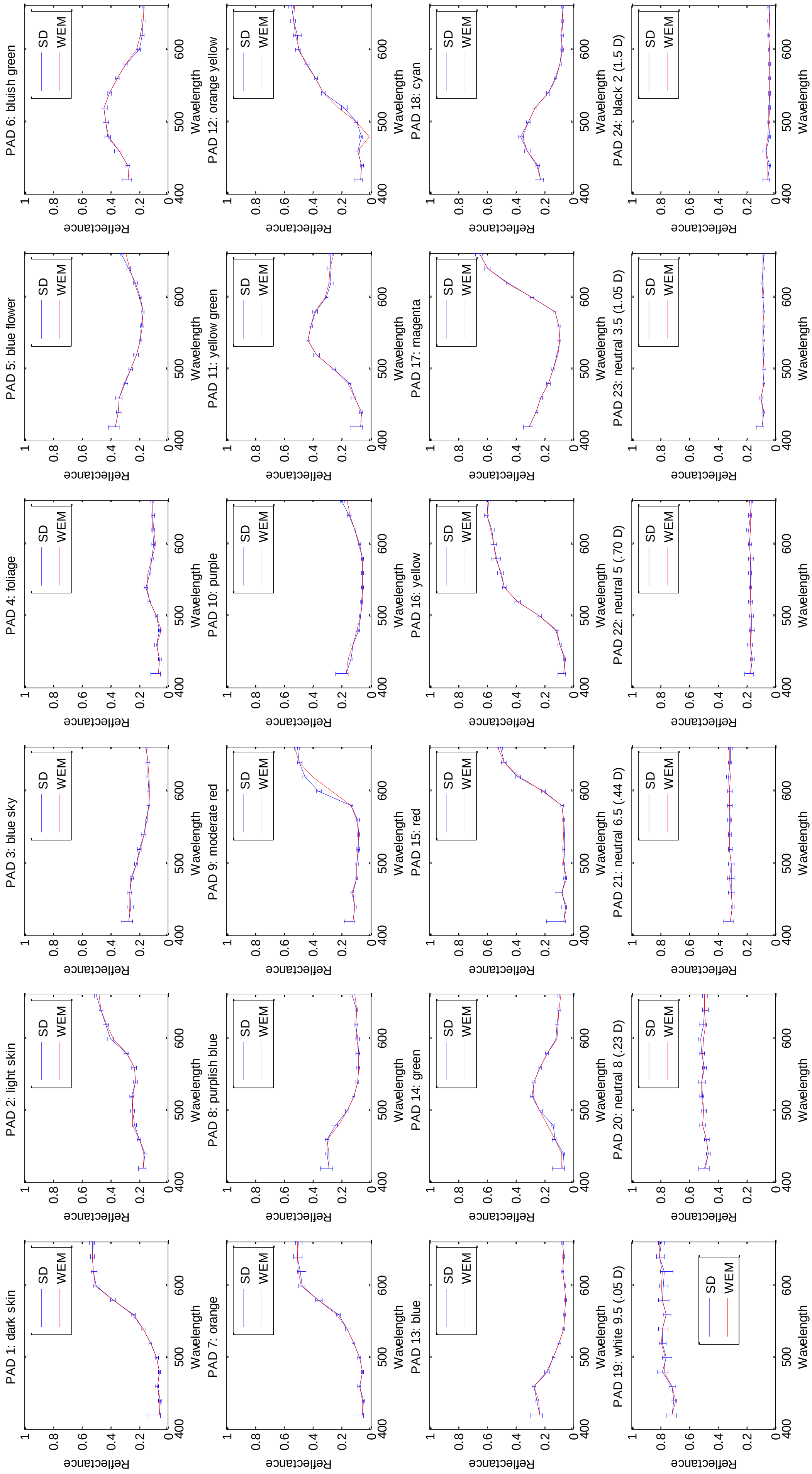


	Wiener Estimation RGB & NB 2 (no 2^{nd} ROW)			Thresholds		
Pad	Euclidian Norm	GFC	Spectral Angle	One	Two	Three
1	0,05743	0,999204	0,039915	1	1	1
2	0,07604	0,998827	0,048434	1	1	0
3	0,035588	0,998837	0,048236	1	1	0
4	0,053399	0,996873	0,079108	1	1	0
5	0,039061	0,999336	0,036433	1	1	1
6	0,042274	0,999359	0,0358	1	1	1
7	0,412932	0,980023	0,200218	0	0	0
8	0,133936	0,982885	0,185276	0	0	0
9	0,141744	0,99403	0,10932	1	0	0
10	0,170458	0,964454	0,267427	0	0	0
11	0,062609	0,998671	0,051559	1	1	0
12	0,082598	0,999062	0,043325	1	1	1
13	0,06763	0,993295	0,115869	1	0	0
14	0,032023	0,998832	0,048328	1	1	0
15	0,063792	0,99853	0,054236	1	1	0
16	0,032541	0,99983	0,018436	1	1	1
17	0,033053	0,999762	0,021803	1	1	1
18	0,028621	0,999317	0,036954	1	1	1
19	0,056834	0,999834	0,018215	1	1	1
20	0,058454	0,999577	0,029086	1	1	1
21	0,037398	0,999554	0,029861	1	1	1
22	0,025478	0,999259	0,0385	1	1	1
23	0,023532	0,997694	0,067928	1	1	0
24	0,026495	0,986347	0,165437	0	0	0
	Percentage Of Pads With Successful Estimation			83,33%	75,00%	45,83%

		Thresholds		
No	Algorithm	One	Two	Three
1	Wiener RGB	70,83%	58,33%	8,33%
2	Wiener RGB & NB 2(no n2d row)	83,33%	75,00%	45,83%
3	Wiener RGB & NB 2	87,50%	79,17%	33,33%

In this case, we can see that we lost a little on the first two thresholds but we gained significantly on the third one. That means that we estimated accurately less pads, but we had better accuracy with the ones we did estimate. From the plots (Image 51) we can see that we got a very good fit and that in contrast with the use of RGB & NB1 we got good fitting from the pads of the second row that are “unknown”. Thus we saw that we got good results in this case without the use of the seconds row spectra a priori. Next, check was the six Narrow bands input.

Image 49 : Estimation Using Six Narrow Bands - NO 2nd Row



Pad	Euclidian Norm	GFC	Spectral Angle	One	Two	Three
	Wiener Estimation Six Narrow Bands (no 2nd ROW)			Thresholds		
1	0,034829	0,999804	0,019824	1	1	1
2	0,062207	0,999384	0,035105	1	1	1
3	0,021728	0,999602	0,028232	1	1	1
4	0,037196	0,995947	0,090062	1	1	0
5	0,070982	0,998052	0,062433	1	1	0
6	0,055511	0,998937	0,04611	1	1	0
7	0,041001	0,999821	0,018907	1	1	1
8	0,171633	0,974198	0,227654	0	0	0
9	0,129013	0,994462	0,105294	1	0	0
10	0,248854	0,930155	0,375962	0	0	0
11	0,063756	0,998474	0,055252	1	1	0
12	0,078296	0,998637	0,052224	1	1	0
13	0,015272	0,999594	0,028501	1	1	1
14	0,061593	0,995358	0,096391	1	1	0
15	0,066507	0,99905	0,043595	1	1	1
16	0,016414	0,999961	0,008886	1	1	1
17	0,031946	0,999833	0,01826	1	1	1
18	0,042884	0,998442	0,055835	1	1	0
19	0,045724	0,999901	0,014097	1	1	1
20	0,050994	0,99965	0,026445	1	1	1
21	0,037185	0,999566	0,029464	1	1	1
22	0,025685	0,999383	0,035127	1	1	1
23	0,013148	0,999229	0,039276	1	1	1
24	0,018827	0,99465	0,103482	1	0	0
	Percentage Of Pads With Successful Estimation			91,67%	83,33%	54,17%

		Thresholds		
No	Algorithm	One	Two	Three
1	Wiener RGB	70,83%	58,33%	8,33%
2	Wiener Six Narrow Bands (no 2nd row)	91,66%	87,50%	54,16%
3	Wiener Six Narrow Bands	91,67%	83,33%	50,00%

In the final step we saw that we had better results. We got a small rise on the second and third threshold which meant that we got the same number of pads estimated but we got them with better accuracy than before. We could also see that we got very good fitting in the plots (Image 52) for the second row which meant that we were able to estimate accurately "unknown" elements that we didn't have any spectra information for them a priori.

Image 50 : HyperSpectral Image – Multispectral Estimation With RGB (No 2nd row) D% 16.52

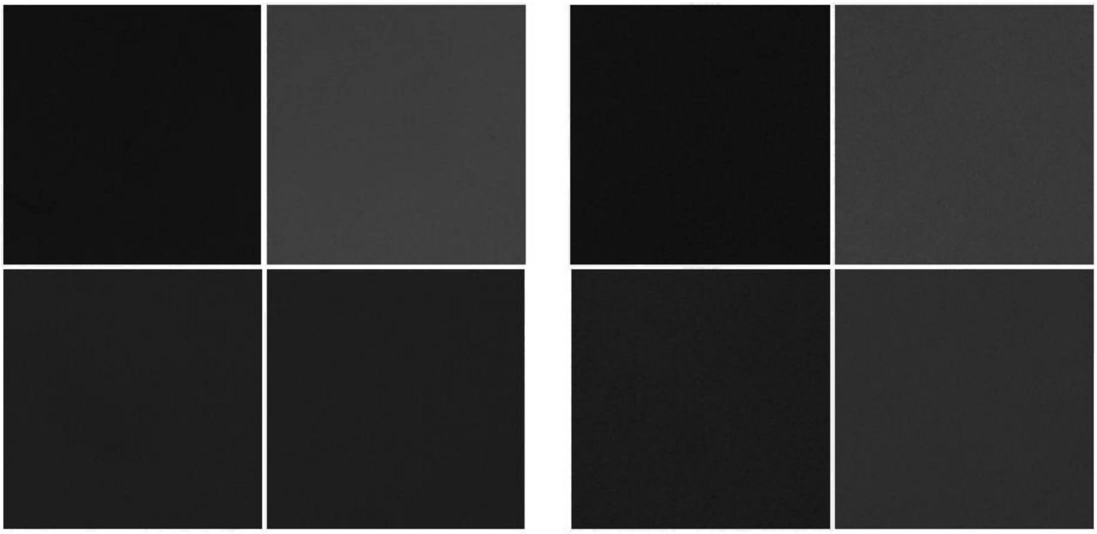


Image 51 : HyperSpectral Image – Multispectral Estimation With RGB & NB 1(No 2nd row)

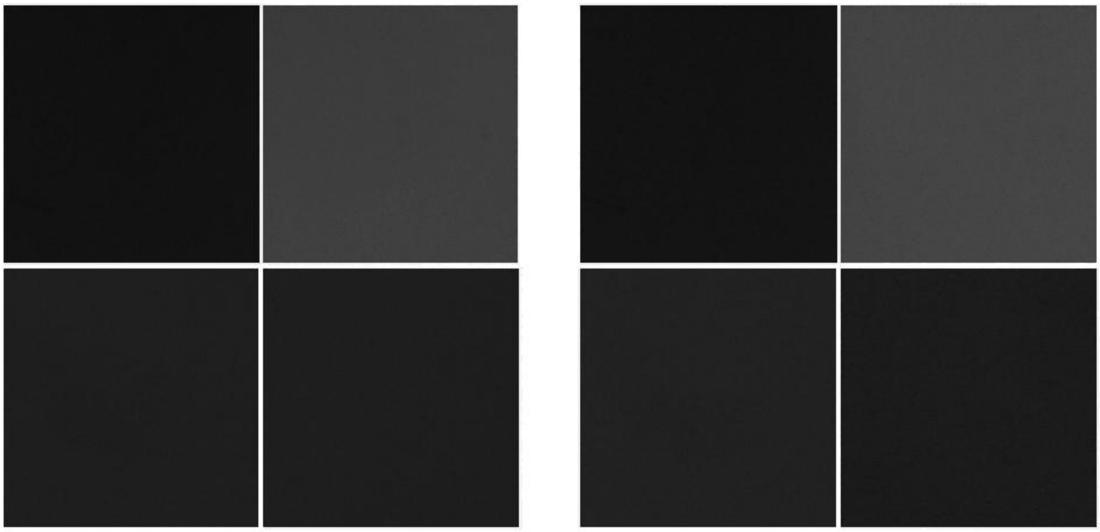


Image 52: HyperSpectral Image – Multispectral Estimation With RGB & NB 2(No 2nd row)

Image 53 : HyperSpectral Image – Multispectral Estimation With Six Narrow Bands (No 2nd row) D% 3.51

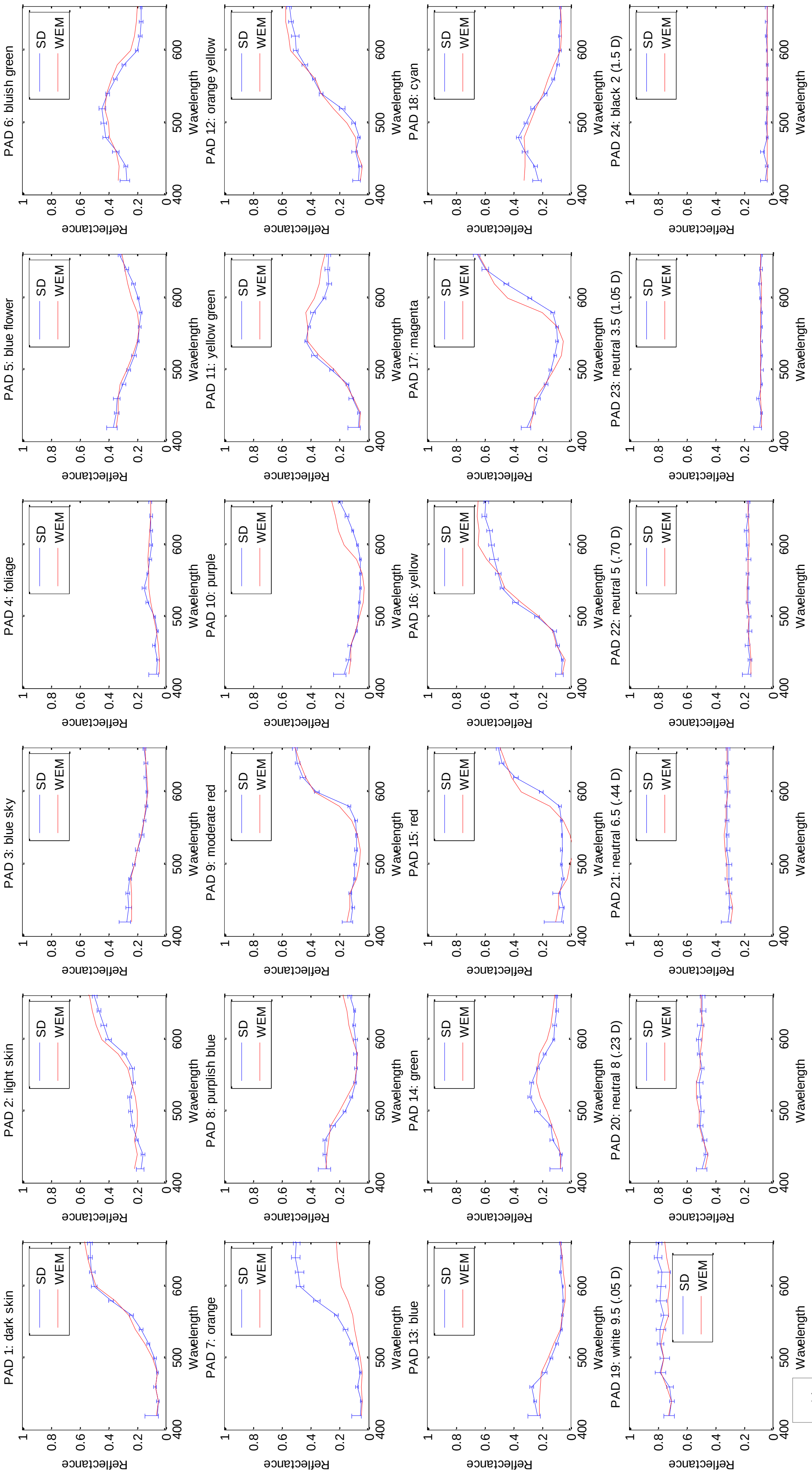
For RGB & NB 1 (Image 54) we had for pad1 1.83%, for pad2 7.12%, for pad7 11.44% and finally for pad8 33%. So as we can see we can got even better results than the ones from full data knowledge. We saw that only the pads of the second row didn't perform that well because we didn't have any a priori knowledge for them. For RGB & NB 2 (Image 55) we had for pad1 8.6 %, for pad2 9.66 %, for pad7 8.52% and finally for pad8 9.7%. From that we see that the results were approximately the same with the ones with full a priori knowledge.

Finally with the Six Narrow Bands input we had a small raise of the difference percentage (about 2%). So, again we had the best results with Six Narrow Bands input.

Thus, the same conclusion applies here. We can still make an accurate estimation, especially when using Six Narrow Band Pass Filters.

Next and final step to check was the removal of the row with all the basic and complementary colors (3rd row).

Image 54 : Estimation Using RGB - NO 3rd Row

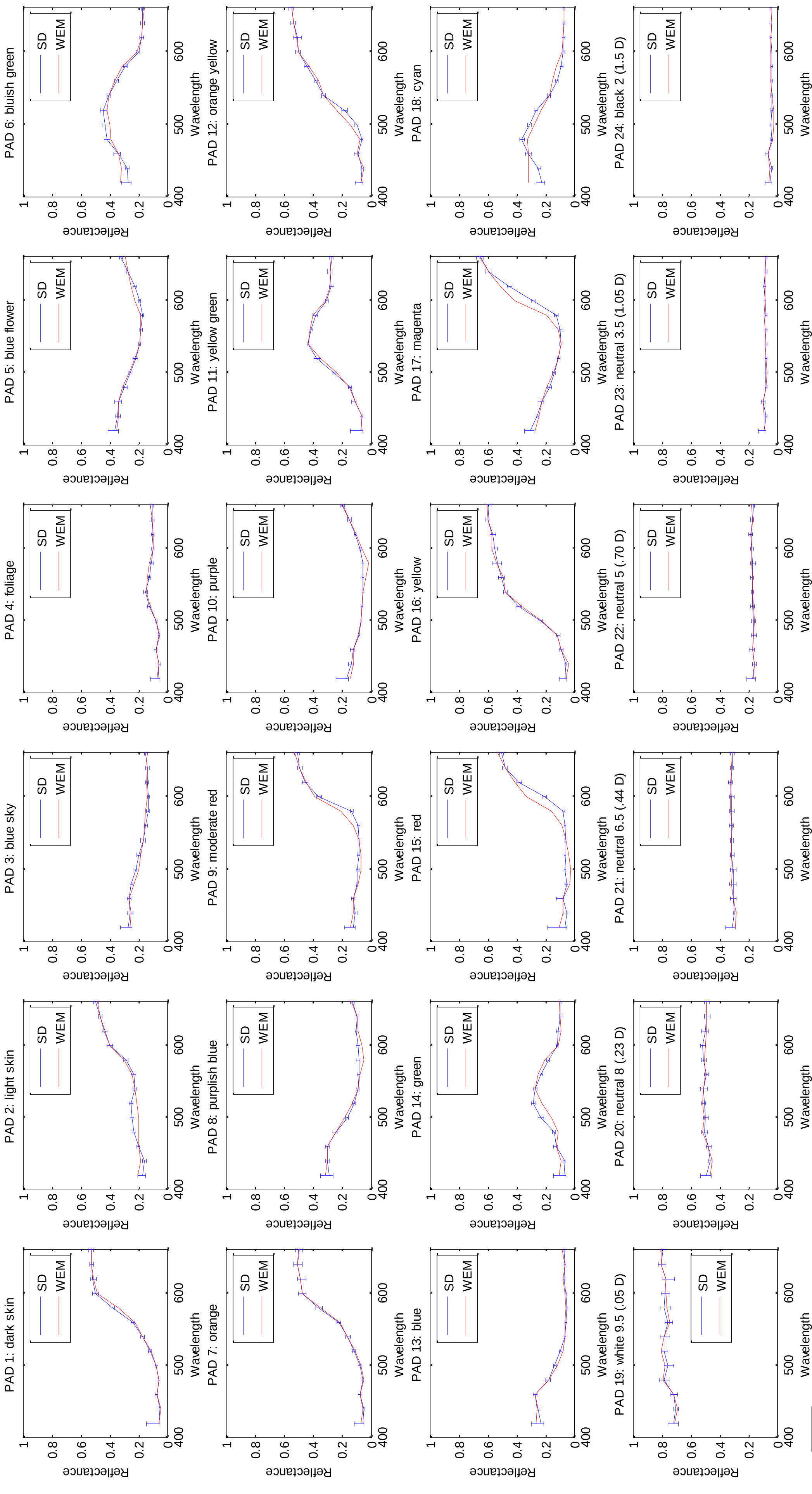


	Wiener Estimation RGB (no 3rd ROW)			Thresholds		
Pad	Euclidian Norm	GFC	Spectral Angle	One	Two	Three
1	0,076299	0,998743	0,050142	1	1	0
2	0,147121	0,996292	0,086138	1	1	0
3	0,051312	0,99858	0,053307	1	1	0
4	0,056472	0,99003	0,141326	1	0	0
5	0,084124	0,997276	0,073829	1	1	0
6	0,134676	0,994348	0,106375	1	0	0
7	0,772705	0,993509	0,114002	1	0	0
8	0,113458	0,988484	0,151908	0	0	0
9	0,123938	0,995875	0,090857	1	1	0
10	0,18664	0,958331	0,289695	0	0	0
11	0,133413	0,994431	0,105586	1	0	0
12	0,136594	0,998257	0,059057	1	1	0
13	0,086143	0,988197	0,153794	0	0	0
14	0,13322	0,979089	0,204863	0	0	0
15	0,212105	0,984342	0,177194	0	0	0
16	0,181709	0,997799	0,066364	1	1	0
17	0,215826	0,992315	0,124052	1	0	0
18	0,138243	0,985531	0,170318	0	0	0
19	0,132869	0,99945	0,033169	1	1	1
20	0,089443	0,999063	0,043294	1	1	1
21	0,064414	0,998782	0,049363	1	1	0
22	0,035838	0,998512	0,054565	1	1	0
23	0,025444	0,996875	0,079076	1	1	0
24	0,030019	0,987427	0,158739	0	0	0
	Percentage Of Pads With Successful Estimation			70,83%	50,00%	8,33%

		Thresholds		
No	Algorithm	One	Two	Three
1	Wiener RGB	70,83%	58,33%	8,33%
2	Wiener RGB (no 3rd row)	70,83%	50,00%	8,33%

From the above table we can see that we had approximately the same results as with full data. The only difference is that we lost some percentage on the second thresholds which means that we got the same number of pads estimated but we didn't got them as accurately as before. Next was the RGB & NB 1 input.

Image 55 : Estimation Using RGB 7 NB1 - NO 3rd Row

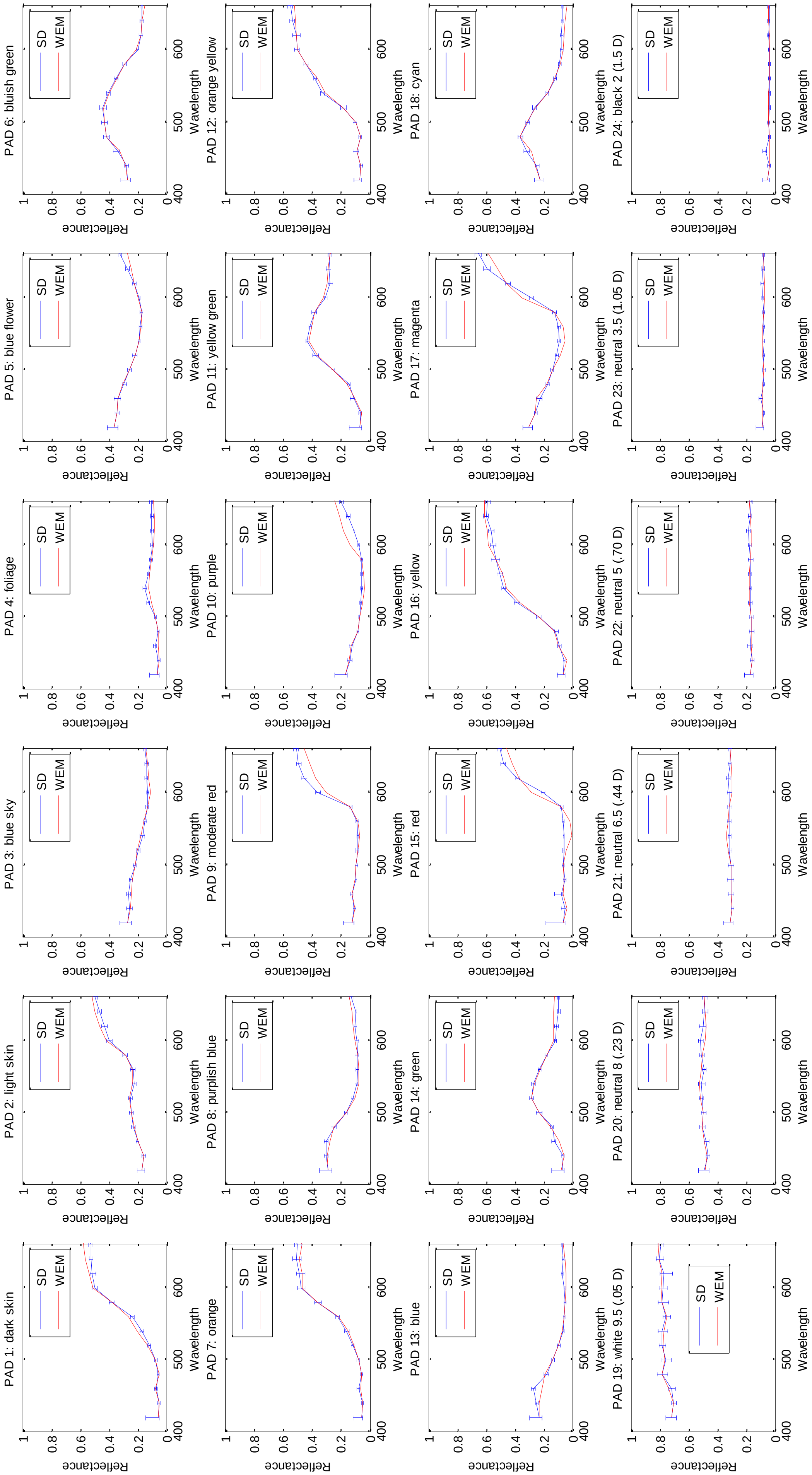


	Wiener Estimation RGB & Triple 1(no 3rd ROW)			Thresholds		
Pad	Euclidian Norm	GFC	Spectral Angle	One	Two	Three
1	0,097598	0,998716	0,050688	1	1	0
2	0,099841	0,997701	0,067827	1	1	0
3	0,04611	0,998115	0,06141	1	1	0
4	0,058244	0,991879	0,127528	1	0	0
5	0,064445	0,998234	0,059437	1	1	0
6	0,120631	0,994773	0,102285	1	0	0
7	0,070868	0,999206	0,039856	1	1	1
8	0,090339	0,99228	0,12434	1	0	0
9	0,139989	0,996163	0,087625	1	1	0
10	0,090682	0,994822	0,101808	1	0	0
11	0,047961	0,999133	0,041652	1	1	1
12	0,092849	0,998315	0,058067	1	1	0
13	0,064272	0,994281	0,107003	1	0	0
14	0,115432	0,983916	0,179593	0	0	0
15	0,239849	0,989283	0,146536	0	0	0
16	0,069536	0,999412	0,034287	1	1	1
17	0,200884	0,994675	0,103243	1	0	0
18	0,14143	0,984179	0,178115	0	0	0
19	0,054511	0,999842	0,017802	1	1	1
20	0,058087	0,999536	0,030456	1	1	1
21	0,035517	0,99957	0,02933	1	1	1
22	0,02878	0,999125	0,04183	1	1	1
23	0,025649	0,99772	0,067547	1	1	0
24	0,039073	0,974953	0,224287	0	0	0
	Percentage Of Pads With Successful Estimation			83,33%	58,33%	29,17%

		Thresholds		
No	Algorithm	One	Two	Three
1	Wiener RGB	70,83%	58,33%	8,33%
2	Wiener RGB & NB 1 (no 3rdrow)	83,33%	58,33%	29,17%
3	Wiener RGB & NB1	83,33%	79,17%	20,83%

In this case we could see that we had the same number of pads estimated correctly, but we lose some percentage on the second threshold which meant that we lost on accuracy. On the other hand we had a rise on the third threshold which meant that this gave us better accuracy. That means that we had better results in some pads and worst results on others. In general from the plots (Image 58) we can see that we have good fitting. Next check was RGB & NB2 input.

Image 56 : Estimation Using RGB & NB2 - NO 3rd Row

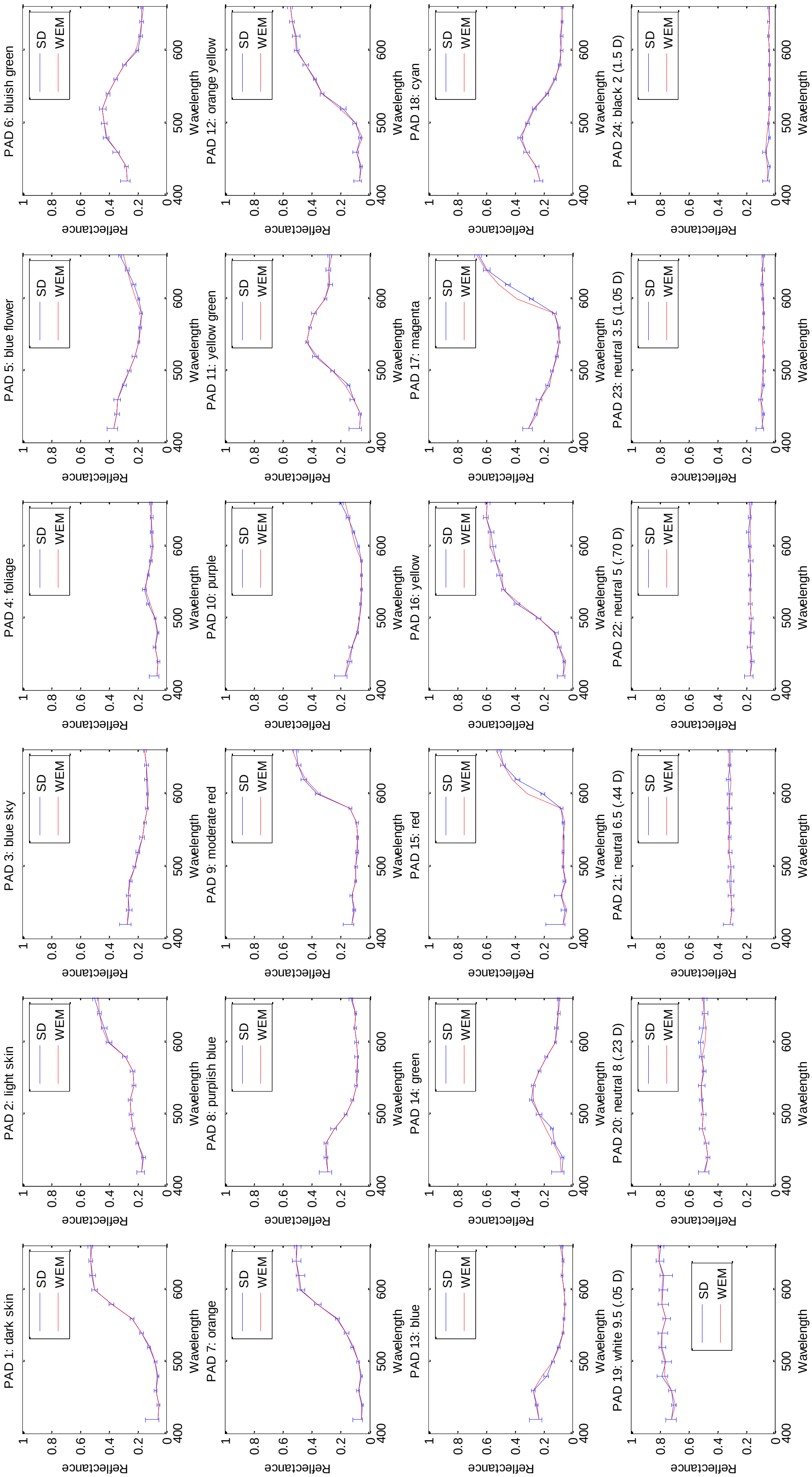


Wiener Estimation RGB & NB 2 (no 3rd ROW)				Thresholds		
Pad	Euclidian Norm	GFC	Spectral Angle	One	Two	Three
1	0,090674	0,999338	0,036393	1	1	1
2	0,062905	0,999613	0,027818	1	1	1
3	0,03873	0,998805	0,048887	1	1	0
4	0,061976	0,995791	0,091786	1	1	0
5	0,073424	0,998072	0,062114	1	1	0
6	0,082943	0,99772	0,067535	1	1	0
7	0,075951	0,999594	0,028498	1	1	1
8	0,106158	0,990412	0,138591	1	0	0
9	0,136075	0,99641	0,084759	1	1	0
10	0,146518	0,972157	0,236531	0	0	0
11	0,073598	0,998006	0,063166	1	1	0
12	0,050656	0,999759	0,021963	1	1	1
13	0,070929	0,993517	0,113932	1	0	0
14	0,066517	0,994616	0,103819	1	0	0
15	0,148039	0,991362	0,131536	1	0	0
16	0,073437	0,999317	0,036962	1	1	1
17	0,144899	0,995997	0,089507	1	1	0
18	0,070104	0,996101	0,088337	1	1	0
19	0,075778	0,999762	0,021802	1	1	1
20	0,075198	0,99927	0,038201	1	1	1
21	0,048277	0,999206	0,039851	1	1	1
22	0,033593	0,998779	0,049414	1	1	0
23	0,027753	0,996967	0,07791	1	1	0
24	0,027603	0,985478	0,170628	0	0	0
	Percentage Of Pads With Successful Estimation			91,67%	75,00%	33,33%

		Thresholds		
No	Algorithm	One	Two	Three
1	Wiener RGB	70,83%	58,33%	8,33%
2	Wiener RGB & NB 2(no 3rd row)	91,67%	75,00%	33,33%
3	Wiener RGB & NB 2	87,50%	79,17%	33,33%

In this case we saw a rise on the first threshold which means that we could estimate accurate more pads than before. We also saw a small drop on the second threshold which again means that we lost a bit of accuracy. In general we see that by omitting the data from the 3rd row we can estimate accurately more pads than before. We can also see in the plots (image 59) that we also have a very good fit. Next and final check was the six band filter.

Image 57: Estimation Using Six Narrow Bands - NO 3rd Row



	Wiener Estimation Six Narrow Bandss (no 3rd ROW)			Thresholds		
Pad	Euclidian Norm	GFC	Spectral Angle	One	Two	Three
1	0,064555	0,999435	0,033618	1	1	1
2	0,095348	0,998292	0,058448	1	1	0
3	0,028364	0,999303	0,037342	1	1	1
4	0,02447	0,998051	0,062438	1	1	0
5	0,041548	0,999237	0,039066	1	1	1
6	0,056338	0,998893	0,047068	1	1	0
7	0,021962	0,999933	0,011587	1	1	1
8	0,077957	0,995176	0,098265	1	1	0
9	0,082936	0,998457	0,055564	1	1	0
10	0,222622	0,945412	0,33194	0	0	0
11	0,087515	0,997235	0,074386	1	1	0
12	0,039679	0,999696	0,024651	1	1	1
13	0,088039	0,988908	0,149082	0	0	0
14	0,076819	0,992752	0,120468	1	0	0
15	0,134931	0,995782	0,091876	1	1	0
16	0,051227	0,99959	0,02863	1	1	1
17	0,117414	0,997372	0,072514	1	1	0
18	0,063683	0,996724	0,080971	1	1	0
19	0,055933	0,999848	0,017448	1	1	1
20	0,070279	0,999339	0,036357	1	1	1
21	0,051288	0,999169	0,040775	1	1	1
22	0,040049	0,998485	0,055059	1	1	0
23	0,030059	0,996707	0,081178	1	1	0
24	0,03148	0,987282	0,159654	0	0	0
	Percentage Of Pads With Successful Estimation			87,50%	83,33%	37,50%

		Thresholds		
No	Algorithm	One	Two	Three
1	Wiener RGB	70,83%	58,33%	8,33%
2	Wiener Six Narrow Bands (no 3rd row)	87,50%	83,33%	37,50%
3	Wiener Six Narrow Bands	91,67%	83,33%	50,00%

In this case we see that we hads a fall on all thresholds. That's a logical drop since the data removed were of the basic colors that are used to produce all others. We could also notice that the percentages of the thresholds were quite high in comparison with others input methods and also that we got very good fit from the plots (Image 60) as we did in all cases with this input before. So, one more time we could see that we could make an accurate estimation of unknown samples using Six Narrow Bands. Final step was to check how this lack of knowledge affects the Spectral Cube production.

Image 58: HyperSpectral Image – Multispectral Estimation With RGB (No 3rd row) D% 10.77%

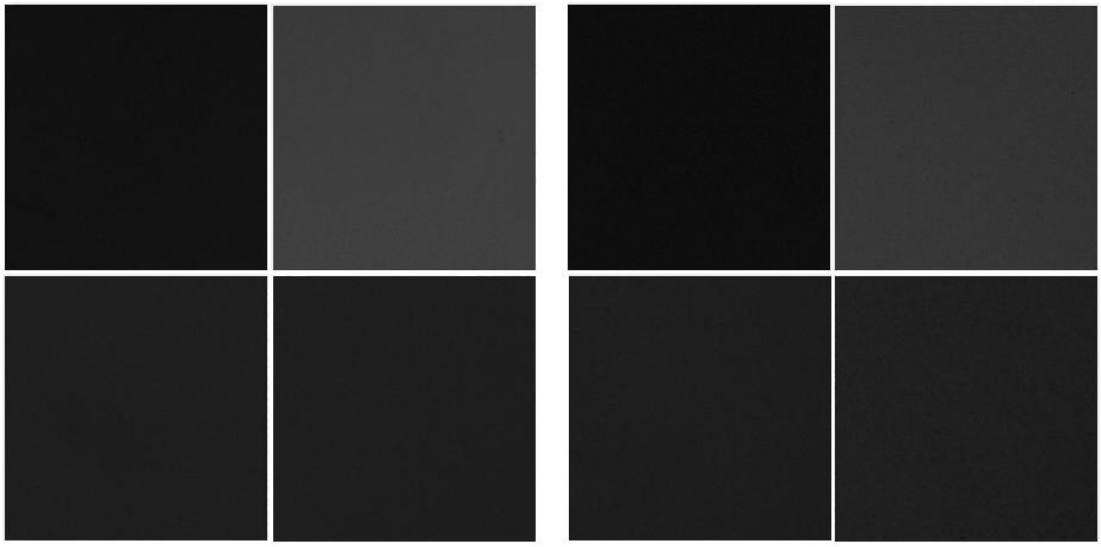


Image 59: HyperSpectral Image – Multispectral Estimation With RGB & NB 1 (No 3rd row)

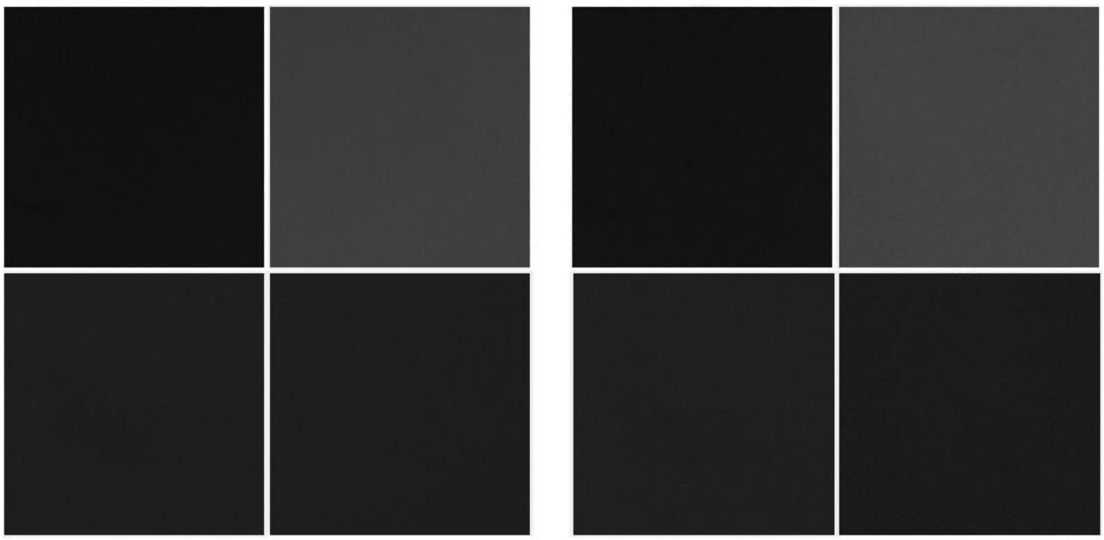


Image 60: HyperSpectral Image – Multispectral Estimation With RGB & NB 2 (No 3rd row) D% 0.41%

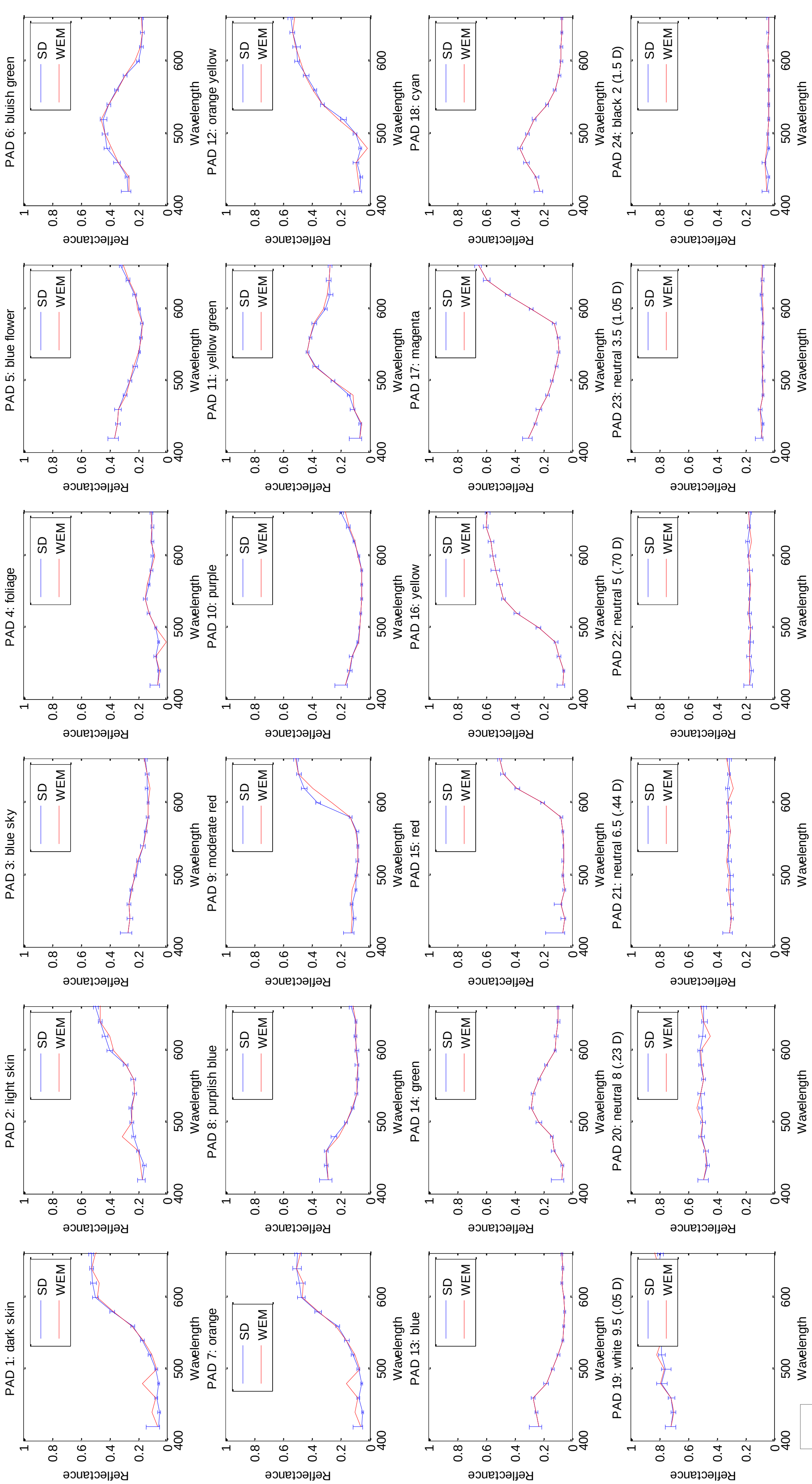
Image 61: HyperSpectral Image – Multispectral Estimation With Six Narrow Bands (No 3rd row) D% 2.09%

For RGB & NB 1 (Image 62) we had for pad1 20.75%, for pad2 18.20%, for pad7 2.77 % and finally for pad8 3.9%. For RGB & NB 2 (Image 63) we had for pad1 7.44 %, for pad2 7.49 %, for pad7 4.18 % and finally for pad8 9.57 %. So, from the above results we could see that in the combination with NB1 we had a significant increase of the difference. On the other hand we saw a decrease on the second row. The main reason that we have very different results from before is because the third row that contains all the basic and complementary colors can be proven very useful since all colors can be got from a combination of the former. For the combination with NB2 we can see a bit better results than with full knowledge. In general, from all the above cases, we saw that the method of RGB & NB2 gives better results over RGB & NB1. For the Six Narrow Bands we can see that in this case we also got the best results.

So, concluding from this section we gpt that the best input method for a spectral estimation is the use of Six Bands and that we could still make an accurate estimation without have all spectral data of a sample a priori.

Next we tried a different approach. We used only the third row in the training set and saw if we could make an accurate estimation on all pads just by having the basic and complementary colors. Since we have seen that we got our best results for Six Narrow Band we will use only this type of data input.

Image 62: Estimation Using Six Narrow Bands - Only 3rd Row



Wiener Estimation Six Narrow Bandss (only 3rd ROW)				Thresholds		
Pad	Euclidian Norm	GFC	Spectral Angle	One	Two	Three
1	0,237374	0,989502	0,145027	0	0	0
2	0,183405	0,992915	0,119112	1	0	0
3	0,050824	0,997711	0,067669	1	1	0
4	0,071247	0,984403	0,17685	0	0	0
5	0,032027	0,999545	0,030184	1	1	1
6	0,051446	0,999056	0,043459	1	1	1
7	0,176672	0,992602	0,121716	1	0	0
8	0,145018	0,982275	0,18856	0	0	0
9	0,117943	0,995492	0,094986	1	1	0
10	0,184751	0,968168	0,252992	0	0	0
11	0,043143	0,999314	0,03704	1	1	1
12	0,082587	0,998582	0,053257	1	1	0
13	2,32E-14	1	2,11E-08	1	1	1
14	2,59E-14	1	0	1	1	1
15	9,28E-15	1	2,11E-08	1	1	1
16	3,85E-14	1	0	1	1	1
17	2,21E-14	1	2,11E-08	1	1	1
18	4,35E-14	1	0	1	1	1
19	0,127282	0,999276	0,038045	1	1	1
20	0,130156	0,997923	0,064463	1	1	0
21	0,087226	0,997724	0,067476	1	1	0
22	0,04639	0,997946	0,064108	1	1	0
23	0,02026	0,998261	0,058978	1	1	0
24	0,021899	0,991729	0,128704	1	0	0
	Percentage Of Pads With Successful Estimation			83,33%	70,83%	41,67%

		Thresholds		
No	Algorithm	One	Two	Three
1	Wiener RGB	70,83%	58,33%	8,33%
2	Wiener Six Narrow Bands (only 3rd row)	83,33%	70,83%	41,67%
3	Wiener Six Narrow Bands	91,67%	83,33%	50,00%

Here we can see that we had a drop on all thresholds. But as we can see from the plots and the threshold table we got for the first time 100% match of the measured and estimated spectra. We saw that all the 3rd row has perfect fit. Also we can see that we have descent results for all pads. That can be seen from the plots and from the threshold statistics which were still pretty high in comparison with other methods. This experiment just proved that we can have an accurate estimation of all MacBeth by just using the row with the Basic and Complementary color. Next, we present the results on the spectral cube.

Image 63: Six Narrow Bands Estimation (Only 3rd row)(Measured- Estimated) D% 33.57%

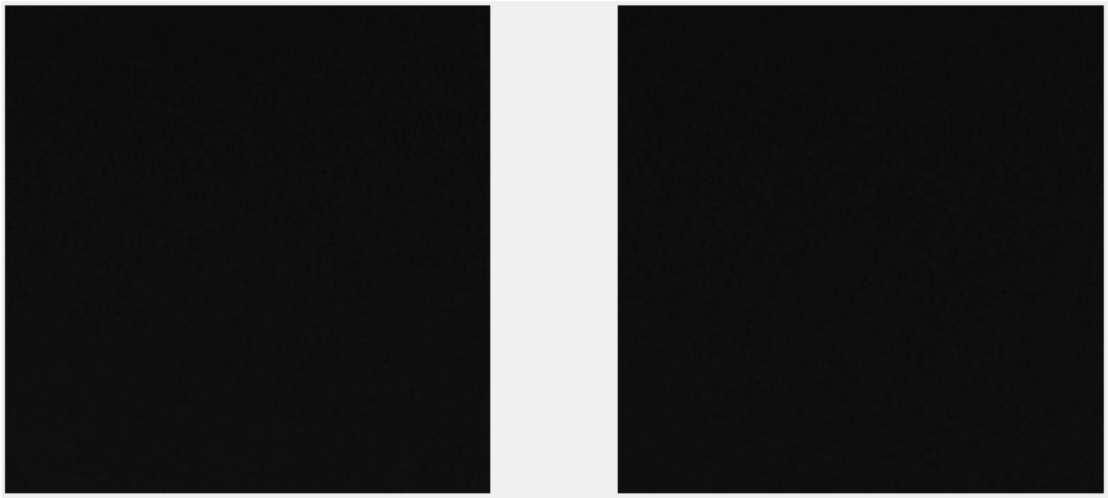


Image 64: Six Narrow Bands Estimation (Only 3rd row)(Measured- Estimated) D% 2.48% Pad16

From the above pictures we can see that in the picture with four pads we have a rise on the difference percentage at 33.57%. furthermore we see that if we choose to estimate a single pad we get a raise at 3.02%. As we can see the raise on the single pad is minor so we could say that we can make an accurate estimate for the cube of each pad separately.

4.7 <u>Unknown Object Estimation</u>

Next, for the end of our research we checked the spectral estimation method using Six Narrow Bands and with full a priori Macbeth data, to estimate a completely unknown object. It will be a canvas with oil paints. From this we wanted to see if we could get a universality out of our procedure so we would be able to train our algorithm with specific data and then be able to estimate anything that we want.

The colors we checked had the following form:

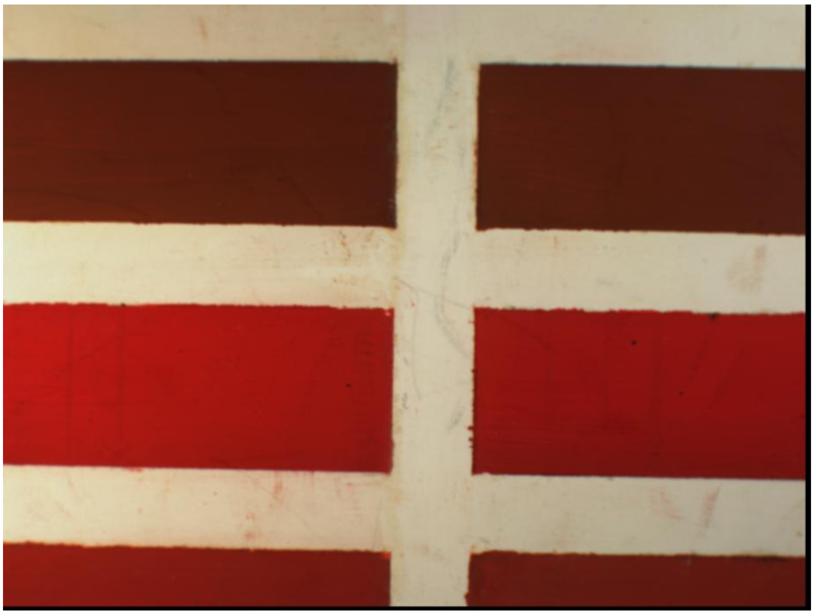


Image 65 : Colors From Unknown Object

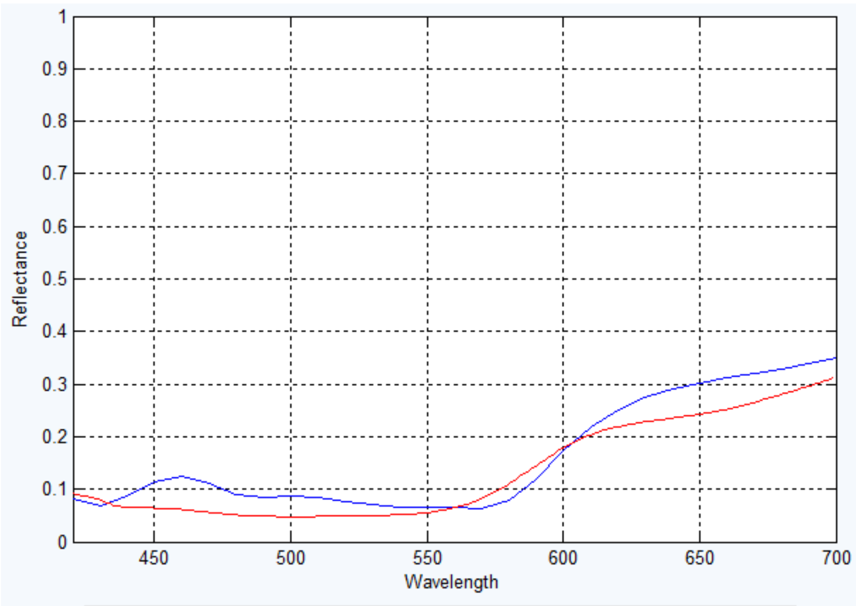


Image 66 : Estimation Of the First top left pad of the picture (Red line : Measured)

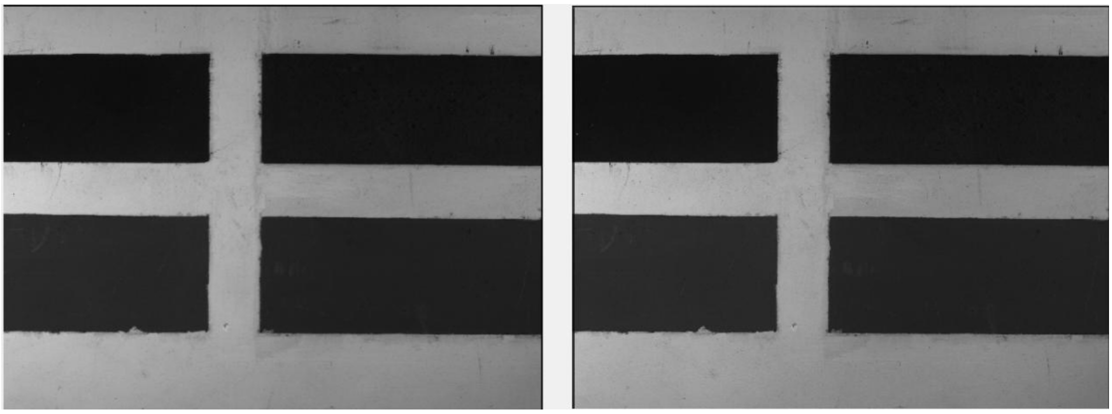


Image 67 : Six Narrow Bands (Measured- Estimated) D% 3.18%

From the result provided above (Images 69-70) we can see that although we don’t have any a priori data for this pads we got a fairly good estimation as it can be seen on the plot and we also get a good Spectral Cube estimation which only differs from the original about 3%. The reason for this percentage is the white frame around the pads. If we look closely on the image of the Spectral Cube we can see that the pads almost have the same color, in contrast with the white which is obviously different in this case. So we must also take in mind that the frame of each sample we want to estimate has a part on the estimation unless we remove it.

Furthermore, we thought that if we could get some of the pads of the new sample we might get better estimation since there will be something familiar in the training set. This experiment is left to be tried in future work.

Chapter 5 : Conclusions and Future Work

Throughout this thesis we studied the methods of Hyper Spectral Estimation Using Filtered CCD Cameras. The original studies made about Spectral Estimation, had as purpose Color Reproduction. When we have the Spectral Information of a color we can reproduce it with better quality. So, they tried to estimate the spectra of paintings using only the RGB values of the camera so they could make a proper reproduction of color using these spectral information.

Also, we have seen that in previous studies they used different cameras to acquire the Spectral and RGB information needed. The first big innovation we made is that with MUSIS HS we could have both information from one device. This is very important since different devices have different calibration and don't give instantly matching results to be used. So, with the use of MUSIS HS we can acquire all spectral and RGB information needed along with the spectral cube of the sample.

In our case, we are very interested in the spectral information we can produce, since these information can be used for various purposes as we have seen. So, we wanted to have high accuracy on our estimations. First step was to validate the results for the algorithms provided. From there, we saw that we can have a spectral estimation, but not with the accuracy wanted. Then, we changed the type of input of the algorithms. We discovered that with the use of Six Band Pass Filters in general we got very good results. Furthermore we saw that with the use of Six Narrow Bands we got the best results with the use of Wiener Estimation Method, who was the best algorithm we chose.

After choosing the best method of input and algorithm we created a method (GUI) that can produce spectral cube of the estimated spectra. We observed that the estimated cube of the Six Narrow Band Pass Filters input is very similar to the original spectral Cube. That means that we can also produce the whole spectral cube of an object by using only Six Bands as inputs.

The accurate estimation of spectra –Spectral Cube means that we have very good information about the spectrum of an object-scene. As, we mentioned before, when we have spectral information we can make an very good color reproduction and handle the phenomenon of the metamerism. That means that by improving the results of the algorithms, we can make color reproduction that will get us better quality of the colors than before when only RGB input was used.

Another innovation we had was to minimize the a priori knowledge needed. In our case the a priori knowledge used was the Spectral and RGB values of the MacBeth Color Checker. That showed us that we can still accurately estimate objects for which we don't have any spectral information for. We also tried to make an estimation using MacBeth as a priori knowledge of a completely unknown object from MacBeth. We used a wooden pad with oil paints. From the results provided we saw that we can make an accurate estimation of an object without knowing any spectral information as a priori knowledge besides the input. We saw that we could also create the estimated spectral cube of the object which also gave good difference results from the original. A change that could provide better results is to use some of the wooden pads spectral information a priori. The important step to be made is to create a universal training set so we could make an estimate about any object provided without the need of having any spectral information a priori for it or create a experiment related training set each time, so we could have spectra that are relative to the experiment in progress.

Also, we know that there are devices that can make snapshot spectral imaging using a certain number of bands and then offline make calculation to complete the spectral cube. Since out algorithm needs very little time for the process time we can create a system that will have Six Band Pass Filters and then have online processing to create the Spectral Cube in Real-time and that can also be used for mobile applications (Since spectral imaging nowadays is mostly for static applications). This is extremely hopeful since in the present moment there is no hyper-spectral imaging device that can provide real time results.

Also, another important thought that needs to be tested is the estimation of spectrum beyond 700nm. That means that we want to estimate, not only the visible spectrum, but also the IR. From there we can seriously get more important information about the tested object. We could also investigate the number of bands that can be used. For example, 9 bands maybe needed in order to make an accurate estimate on both visible and IR spectrum.

Concluding, we now have an accurate estimation method that can provide us with spectral information – cubes about an object by using six bands of the spectrum as input and we also have the ability to estimate spectra of an unknown object without having any a priori spectral knowledge about it. Furthermore we saw that a Real Time Spectral Imaging device can be created which we will have Six Acquisition bands and online processing. Finally, we have the possibility to expand the spectral estimation procedure to IR spectrum.

References

[1] Costas Balas, George Epitropou, Christos Pappas "Multi/Hyper Spectral Imaging"

[2] Georgios N. Stamatas, Costas J. Balas and Nikiforos Kollias, "Hyperspectral image acquisition and analysis of skin"

[3] Daisuke Nakao, Norimichi Tsumura , Yoichi Miyake, "Real-time multi-spectral image processing for mapping pigmentation in human skin"

[4] Konstantinos Rapantzikos, Costa Balas, "HyperSpectral Imaging : Potential in non-destructive analysis of palimpsests"

[5] Abdelhameed Ibrahim , Shoji Tominaga, Takahiko Horiuchi, "Unsupervised Material Classification of Printed Circuit Boards Using Dimension-Reduced Spectral Information"

[6] George Epitropou, "HyperSpectral imaging and spectral classification algorithms for the non-destructive analysis of El Greco's paintings"

[7] Hideaki Haneishi, Takayuki Hasegawa, Asako Hosoi, Yasyaki Yakoyama , Norimichi Tsumura and Yoichi Miyake, " System design for accurately estimating the spectral reflectance of art painting"

[8] P. Stigell, K.Miyata and M. Hauta – Kasari, "Wiener Estimation Method in Estimating of Spectral Reflectance from RGB Images." (2005)

[9] Francisco H. Imai , Mitchell R. Rosen , Roy S. Berns, "Comparison of Spectrally Narrow-Band Capture Versus Wide-Band with a Priori Sample Analysis for Spectral Reflectance Estimation "(2000)

[10] Alamin Mansouri, Tadeusz Sliwa, Jon Y. Hardeberg, Yvon Voisin, "Representation and Estimation Of Spectral Reflectances Using Projection on PCA and Wavelet Bases" (2008)

[11] Alamin Mansouri, Tadeusz Sliwa, Jon Y. Hardeberg, Yvon Voisin, "An Adaptive-PCA Algorithm for Reflectance Estimation from Color Images" (2008)

[12] Francisco H. Imai , Roy S. Berns and Di-Y. Tzeng , "A comparative Analysis of Spectral Reflectances Estimated in Various Spaces Using a trichromatic Camera System"

[13] Mathias Johansson, " The Hilbert Transform"

[14] Wall, Michael E., Andreas Rechtsteiner, Luis M. Rocha."Singular value decomposition and principal component analysis". in A Practical Approach to Microarray Data Analysis. D.P. Berrar, W. Dubitzky, M. Granzow, eds. pp. 91-109, Kluwer: Norwell, MA (2003).

[15] Ahmed, N.; Natarajan, T.; Rao, K.R."Discrete Cosine Transform"

[16] Hans Riedwyl , "Goodness Of Fit" (1967) Jolliffe I.T. "Principal Component Analysis, Series: Springer Series in Statistics", 2nd ed., Springer, NY, 2002, XXIX, 487 p. 28

[17] Francisco H. Imai , Mitchell R. Rosen and Roy S. Berns, "Comparative Study of Metrics for Spectral Match Quality" (2002)

[18] N. Shimano and M. Hironaga "Recovery of spectral reflectances of imaged objects by the use of features of spectral reflectances" (2010)

[19] Eva M. Valero, Juan L. Nieves, Sérgio M. C. Nascimento, Kinjiro Amano, David H. Foster , "Recovering spectral data from natural scenes with an RGB digital camera and colored filters"

[20] Hui-Liang Shen, Chen-Wei Weng, Hui-Jiang Wan, and John H. Xin, "Correcting cross-media instrument metamerism for reflectance estimation in multispectral imaging"

[21] Jolliffe I.T. "Principal Component Analysis, Series: Springer Series in Statistics", 2nd ed., Springer, NY, 2002, XXIX, 487 p. 28

[22] Brigham, E. O.; Morrow, R. E. , "The Fast Fourier Transform"

[23] *D.Anglos, C.Balas, C.Fotakis, "Laser spectroscopic and optical imaging techniques in chemical and structural diagnosis of painted artworks" ,Am.Lab.(October)(1999)60–67.*

[24] *C.Balas, "An imaging method and apparatus for the non-destructive analysis of paintings and monuments",* International PatentApp. PCT/GR00/00039.

[25] *C.Balas, "A novel optical imaging method for the early detection, quantitative grading and mapping of cancerous and precancerous lesions of cervix", IEEE Trans.* Biomed. Eng. 48(2001)96–104.