

Effectiveness of a Machine Learning Algorithm and four Narrow Band Sensors Facing twelve Directions for Predicting the Ambient Spectrum

Overview

Author: Hsiumin Wu (AUC) hsiu-min.wu@student.auc.nl

Mentor: Imme Schuringa (AMOLF) I.Schuringa@amolf.nl

Supervisor: Forrest Bradbury (AUC) f.r.bradbury@auc.nl

Co-Supervisor: Bruno Ehrler (AMOLF) b.ehrler@amolf.nl

Reader: Katerina Newell (UvA) k.newell@uva.nl

Tutor: Breannán Ó Nualláin (AUC) b.s.onuallain@uva.nl

Submission Date: 02/06/2021

Major: Sciences

Keywords: Diffuse light detector, Neural Network, Machine Learning, Photovoltaics, Bi-facial

Abstract: There is a perceived need to measure diffuse light in all directions to estimate optimal positions for solar cells, especially bi-facial ones. A 12-faced sensor known as the LAD has been made at AMOLF to measure diffuse light. It measures the light intensity at four wavelengths, red, blue, green and IR, in twelve directions. A single-layer neural network, referred to as NN later, has been made in a previous study which can predict an entire spectrum using the raw spectral data obtained from the LAD. The validity of the predictions, however, is uncertain as the model is trained in only one direction. To validate the model, a spectrometer was used to collect reference spectra from February to April 2021 were collected and compared to corresponding predicted spectra. Further analyses

was done to characterize the prediction's effectiveness. This was done by considering the accuracy of estimating total intensities, the spectral noise and deviancy of shape. We show that the NN works for 8 directions, and less well for the others. Unfortunately the output is not precise enough to determine optimal orientations. This could be due to complex environmental settings, insensitive spectrometer and mismatching angles of incidence between the former and the LAD. Hence further work should be done in an open environment with more attuned equipment.

Introduction

Solar energy holds great promises for energy sustainability due to its accessibility and abundance¹. One of the more advanced methods of extracting this energy, the Photovoltaic module (i.e. PV, colloquially called solar panels), has demonstrated its capabilities and growing potential. The International Energy Agency found that since 2017 PV has continuously contributed more than 50% of the global annual net energy capacity growth from renewable energy sources¹. The same report expected this trend to continue well into 2025¹. With this high demand of PV, it is natural that there is significant interest in its performance. While there exists a wide variation of PVs on the market, crystalline-Silicon cells (c-Si) dominates it with a staggering 95% share of production². As such the focus of such performance issues are logically devoted primarily to c-Si PVs. This type of PV can be categorized into either Mono-facial or bi-facial based on whether the module can use Sunlight for energy production on one side or both.

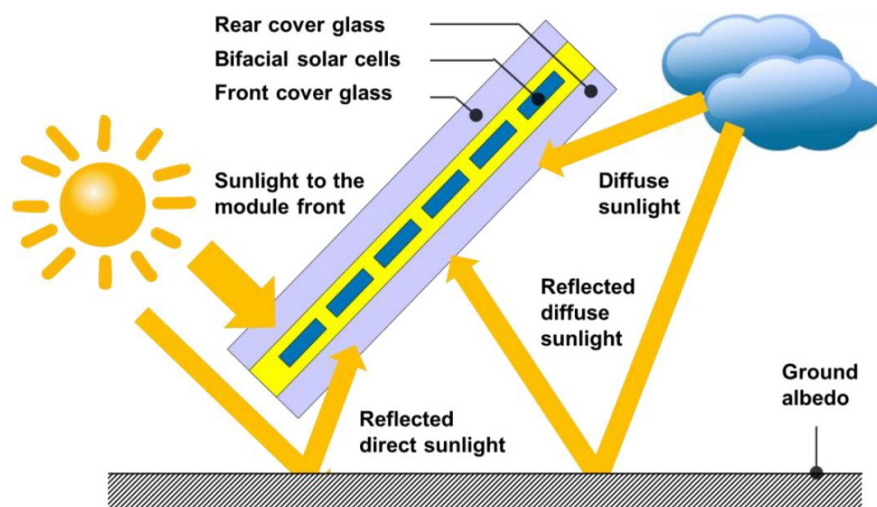


Figure 1: Image indicating the contributions of different types of sunlight to bifacial PVs³.

Of the aforementioned two types of PVs, bi-facial modules have higher energy outputs on average. bi-facial c-Si PVs has been demonstrated to have a higher proportional energy generation than its Mono-facial counterpart by more than 10%⁴. However, unlike mono-facials, bi-facial PVs use primarily diffuse light in one direction, if the said panels are

tilted as to get the other side facing the Sun. The issue with this is that light diffuseness is one of the main factors that impact a PV's performance ratio i.e. its ability to perform in outdoors compared to the calculated yield under the Standard Test Conditions. If the incidence angle is too large, the light would simply be reflected off the panel and not contribute to electricity generation. This is even more important for bi-facial PVs since their backsides are exposed to primarily diffuse light. To maximize energy generation, it is important therefore to keep track of not only the Sun, but also the change of diffuse sunlight in all directions. However, there are limited conclusive studies done on this subject due to the lack of commercially viable measurement tools. Existing tools such as the Solys 2 Sun tracker costs around 10 thousand euros per installation⁵. It follows then that there is a need for the measuring of diffuse light in all directions.

Research Context

AMOLF hosts one of the focal points of the drive for diffuse light measurement tools. It houses a fully operational outdoor Solar Field with multiple commercially viable PVs gathering data year-round. Interest into the light diffuseness factor was first shown with Maassen's⁶ investigation to the aforementioned factor's impact on thin film solar cells. For his investigation a cubic illuminance meter was developed to measure diffuseness. It demonstrated its potential by having a $R^2 > 0.5$ (the coefficient of determination varies from $R^2 = 0$ for no predictive ability to $R^2 = 1$ for perfect predictions) when comparing its measurements with the Solys 2 Sun tracker⁶. This tool was further developed by Kersten⁷ from a cube into a dodecahedron (12 sides) to improve its accuracy and directional scope. This sensor, known by now as the Light Ambient Detector (LAD), only measures in four narrow wavelength bands⁷ centered at: 465nm (Blue), 525nm (Green), 615nm (Red) and 850nm (Infrared). It differs from the usual pyranometers used for sun tracking, which measure the power summed over the entire spectrum via the amount of heat the sensor experiences due to exposure from sunlight⁶. In addition the LAD measures all directions simultaneously, unlike pyranometers which follows the sun. Despite

the limitations in the range of wavelengths it can detect, the LAD has been utilized successfully by Pollastri⁸ to model the cost efficiencies of Bi-facial and Mono-facial Silicon PVs. Preliminary results⁸ suggest that the former has a lower levelized cost of electricity than the latter by 0.5%. To improve on the aforementioned wavelength limitations a Machine Learning Model⁹ has been created to predict a continuous spectrum from the aforementioned LAD inputs; the goal is to be able to generate continuous solar spectra for any given directions using only the LAD inputs. The model, in the form of a Neural Network, is able to generate predictions in one direction with a maximum relative error of 20% when the irradiance E_e , in the units Watts per Meter squared, is in the range $E_e > 400 \frac{W}{m^2}$ and 2% if $40 \frac{W}{m^2} > E_e > 400 \frac{W}{m^2}$ ⁹. However, this Network is only trained with LAD data in one direction; the same direction the local PVs on the Solar Field are facing. Thus the effectiveness of the model in the other 11 directions is still unanswered. To resolve this the NN must be tested in its other directions with reference sunlight spectra.

Tools: LAD, Spectrometer & Neural Network

This section introduces the experimental tools, including the spectrum measurement devices and accessories, the neural network, and the LAD itself. The LAD is a 12-sided dodecahedral diffuse light sensor developed by previous AMOLF Bachelor projects⁶⁻⁹. In each of its surfaces it has a sensor that detects solar intensities in four wavelengths: Blue (465nm), Green (525nm), Red (615nm) and Infrared (850nm). In addition the temperature is also recorded by the sensor for future calibrations to remove its effects on intensity measurements. The LAD is covered with a translucent sphere of plastic to protect it from environmental hazards and is attached to the local Solar Field outside of the AMOLF building. One of its sides, called side A, has the same orientation and angle as the Solar Field, facing southwards. A visual representation of it and its location is shown below with figures 2 and 3.

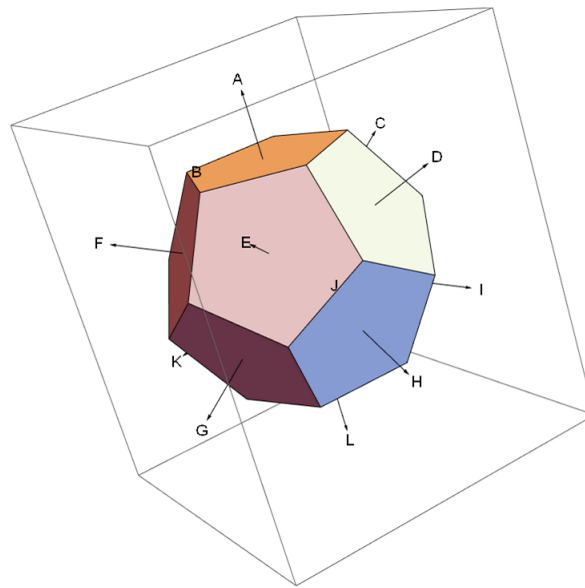


Figure 2: Representation of LAD dodecahedron and Spectrometer Setup dodecahedron. Different directions are marked; direction A faces the sky in the same way as the local Solar Field.



Figure 3: A picture of where the LAD is (the black dodecahedron encased in a translucent sphere in the upper center) and the Solar Field. The metal pole in the left of the LAD is part of the Spectrometer setup as seen later.

The Neural Network is an improvement on the LAD, taking raw LAD values and output entire spectra. It was developed by Blum, a previous AMOLF intern⁹. It is as its name suggests an neural network, a type of Machine Learning Model that simulates

the neurological processes of animals with numbers as the neuron signals and mathematical functions as the synapses that transmit and process signals. For this particular case the Network is trained using LAD values as input and a spectroradiometer, a local spectrometer that faces the same direction as one of the directions of the LAD (direction A), as the desired output. The idea is that the Network was first trained with historical LAD intensities and spectroradiometer spectra that are measured simultaneously as to output the right spectra when given a set of LAD measurements. After the training, this model can now take any LAD data and output a full spectrum. However, since the spectroradiometer is fixed onto the Solar Field, the NN is only trained with LAD values from direction A. Thus it is unclear how the spectra performs in other directions.

The spectrometer is the referential measurement tool we will use to put the NN predictions to test. The test involves using the spectrometer to measure in all the directions the LAD measures to be used as reference data for the NN predictions. It is an AvaSpec-2048-USB2 Spectrometer that can measure light intensities between $\approx 180nm$ to $\approx 1100nm$; exact specifications are available online from its provider Avantes. The spectrometer measures intensities in A/D counts, digital signals that are converted from the voltages the spectrometer gets when light is detected. Being more than ten years old, its calibration is outdated, so a new calibration will have to be done. The spectrometer itself only has a small hole that light can pass through, making it not immediately useable to make measurements comparable to the Predicted spectra coming from LAD measurements in twelve directions. As such a special setup is also required to be installed.

Methodology

spectrometer Setup

The full setup consists of a dodecahedron to measure in 12 directions, the spectrometer and related optical equipment, and the beam support.

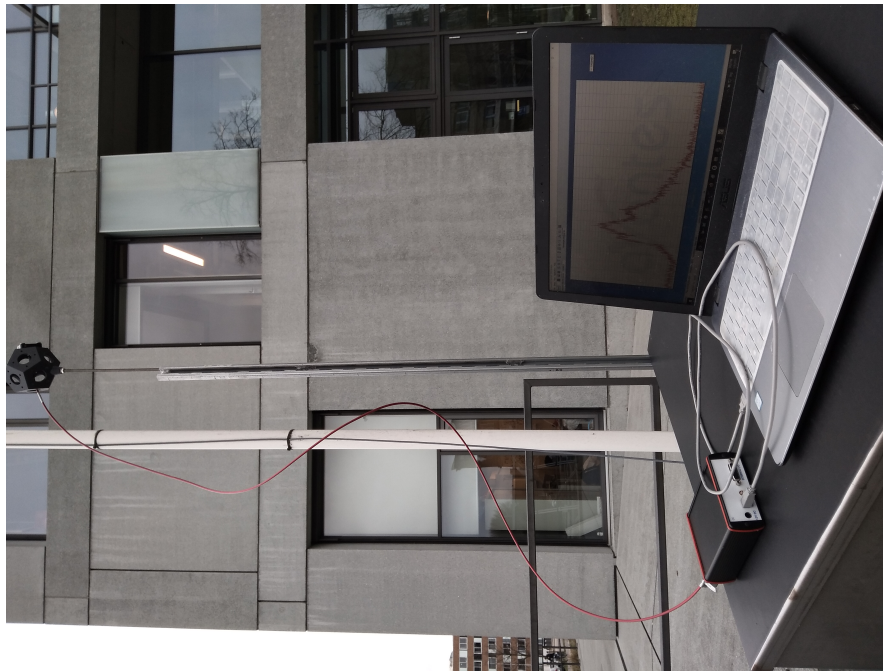


Figure 4: Full spectrometer Setup with Computer example. The platform the computer and spectrometer rests on is the Solar Field Box.

The 3D-printed grey dodecahedron on the top of figure 4 is an imitation of the LAD, such that reference spectra can be obtained in exactly the same directions as the LAD sensor readings. Holes on the object are meant for holding one end of an optical fiber to the face of it for measuring in the precise directions. The dodecahedron's orientation has been calibrated once to be the same as the LAD's, bending the metal bar by 6 ± 0.5 degrees back. The optical fiber is connected to the spectrometer, which will measure the data and send it to the connected computer in real-time. A steel beam is attached to the dodecahedron at one end and a cabinet holding the data processing of the Solar Field at the other. This puts the former at the height of the LAD sensor. This placement is chosen with deliberation. The LAD faces partly the grassy ground and concrete. The current po-

sition, somewhat behind the LAD, has the closest background the LAD would experience.

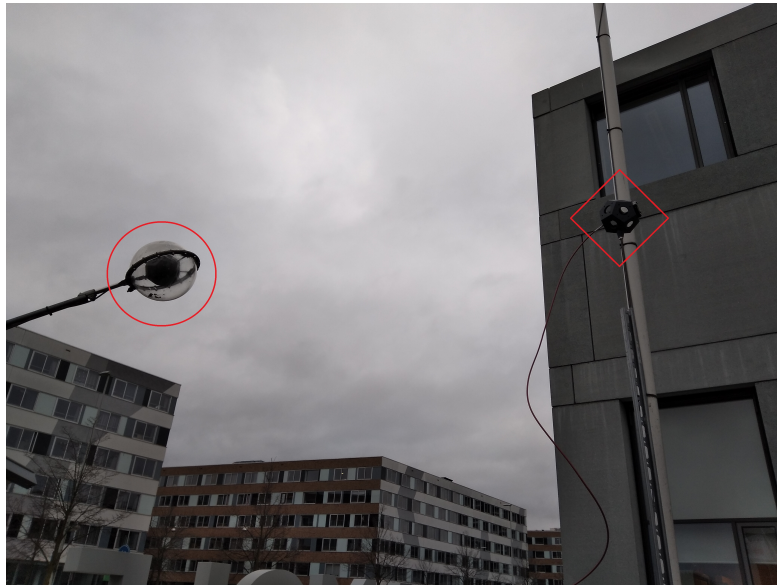


Figure 5: Indication of distance between LAD (marked by Red circle) and spectrometer dodecahedron (marked by red diamond). Picture not representative of actual positioning due to the angle at which it is taken.

To gather light with a wider angle of incidence, the end of the optical fiber that is attached to the dodecahedron has a diffuser on it. This is because the optical fiber has a very small opening that allows little light to enter; this makes the spectrometer very sensitive to slight position changes. The diffuser is a piece of glass that concentrates light passing through it into a point, effectively allowing more light to enter the optical fiber. This is used to make our measurements comparable to the LAD and the spectroradiometer, which both have their own acceptance angles. (Insert acceptance angles table here) The diffuser has an effective acceptance angle of 60 degrees¹⁰.

Measurement Method

spectrometer data is measured in all twelve directions in random times of the day from 9:00 to 17:00. Every direction is measured five times, the time interval between each

measurement being the integration time that is $\leq 500ms$. The position of the optical fiber in the dodecahedron needs to be changed manually. To do this the dodecahedron must first be lowered. After the placeholder with the diffuser and optical fibers have been screwed into place, the contraption is then lifted up again and secured. Afterwards one can hook the optical fiber with the spectrometer and it with a computer to make measurements en masse. Here the semi-automated Avantes software is used to make repeated measurements (five) with a fixed integration speed. Due to breaking of the original diffuser and optical fiber only data from Feb 24th to April 2nd are useable; data collected before mid February is discarded. Since the LAD automatically measures every 6 seconds, one only needs to collect the relevant data from the existing archive. This will be done using Mathematica to find LAD intensities that have the same timestamps as the spectrometer measurements. However, since it is not possible to know at what second spectrometer data is measured, we get all LAD intensities measured during the minute when spectrometer data is measured.

spectrometer Calibration

To ensure that the spectrometer outputs of use for our comparison to the LAD's spectra generated by the Neural Network, calibration must first be done. The spectrometer data are calibrated in two ways: wavelength calibration and Intensity Calibration.

Wavelength Calibration

The wavelength calibration is done to make sure that the irradiance values the spectrometer detects are corresponded to the right wavelength values. To do this the spectrometer is calibrated in the x axis using bandpass filters. Bandpass filters only allow a specific wavelength to pass. Four bandpass filters (560nm, 640nm, 740nm, 810nm) are used to obtain spectrum data. By using them we can tell if there is any shift by finding the peak of the narrow spectra we collected from the filters. Gaussian fits of the data are

then made to obtain the mean wavelengths of each spectrum. These wavelengths are compared to the known Band-pass wavelengths to determine the shift.

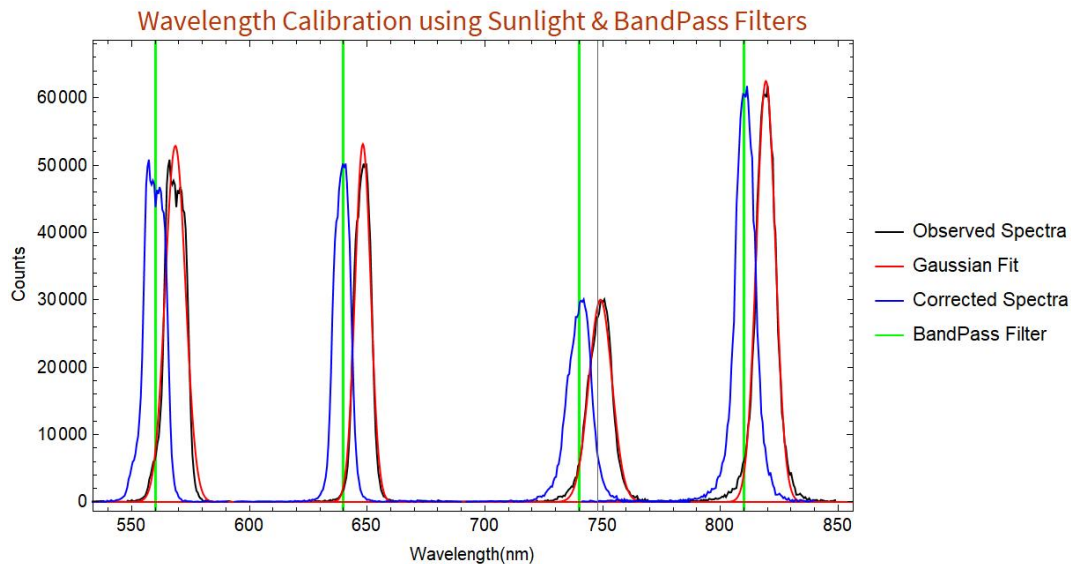


Figure 6: A visual representation of the Wavelength Calibration Process as described. The black curves are measured sunlight spectra using the four mentioned Band-pass filters. The red Gaussian curves are fits to find the maximum peak of the curves. The green lines indicate where the band-pass filter's wavelengths are and thus where the peaks of the spectra should be. The blue curves are spectra corrected with the calculated shifting constant. Generated via Mathematica.

The measurements from figure 6 demonstrated a linear shift of around $10nm$ to the right compared to the band-pass filter lengths. Actual calculations show $8.77075nm$. This constant will then be used to shift the measured spectra to the left with the given length.

Intensity Calibration

There are two goals for the intensity calibration: to correct the spectral intensity into the right amount, and to convert the measured intensities into meaningful physical units. Usually this is done with a light source with a known spectrum in the specified units.

By measuring the spectrum of the light source and correct the spectrometer's output into the given spectrum using the spectrometer's support software, the spectrometer can then measure correctly. However, since such a light source was not available, we resort to correct the spectrometer's output with a existent calibrated spectroradiometer. It is a EKO MS-711 spectroradiometer; the exact specifications can be found on the company (EKO) website¹¹. The spectroradiometer has a field of view of 180 degrees, and is installed on the local Solar Field, facing the same direction as the latter. It is the same one mentioned before that is used to train the Neural Network. It is set to automatically measure spectra from $287.77nm$ to $1121nm$ once every 5 minutes. It outputs spectra in irradiance ($W \cdot m^{-2} \cdot \mu m^{-1}$). The units represent the amount of energy transmitted by sunlight within a given area and for each unit wavelength.



Figure 7: A picture of the spectroradiometer (white disk-like device in the upper middle of the picture), installed on the Solar Field.

As figure 7 suggests, the spectroradiometer is permanently attached to the solar field and calibrated to measure sunlight. As such our intensity calibration is required to also measure under sunlight for calibration. Spectrometer measurements are done using the aforementioned setup.

To convert the spectrometer readings into meaningful units, linear regressions are done

per wavelength. The idea stems from the reasonable assumption that whatever units the spectrometer measures in, it should have a linearly proportional relation with spectroradiometer irradiances. This holds since the spectrometer measures number of photons per wavelength, which is equivalent to an amount of energy per wavelength; a photon of a specific wavelength has a known energy. One can use this relation to make a linear regression line between measured spectrometer intensities in counts and spectroradiometer irradiance for every wavelength value. The result would be a collection of regression lines that will be applied on spectrometer intensities depending on which wavelength the intensity value is associated to.

To obtain the values for creating the linear regressions we make spectrometer measurements on the minute the spectroradiometer measures. To be rid of random errors multiple measurements are made per minute then averaged into one spectrum. These measurements are made in direction A as it is also the spectroradiometer's measurement direction. In total 13 sets of spectrometer measurements are made for this at different times of day, mainly during sunny weather, from February 24th to April 2nd. Since the spectroradiometer does not measure in the same wavelength intervals as the spectrometer, the measured spectrometer spectra are interpolated and from these the measurement sets are converted with comparable wavelength values with spectroradiometer's. spectrometer data are then divided by their integration time such that they are now all per millisecond to make sure the length of exposure when measuring does not vary amongst the intensity values. This does not need to be done for the spectroradiometer as its exposure time is consistently $5000ms$ and irradiance is energy/second (W); the main concern here is consistency within measurement sets by each machine. The spectrometer and spectroradiometer intensities are then extracted per wavelength. Regressions are then made using the function $y = mx$, with the inputs being the spectrometer values and the outputs the spectroradiometer irradiance. A total of 2048 regression lines are generated as per the number of wavelength values it generates. Examples of this are shown in the coming figure:

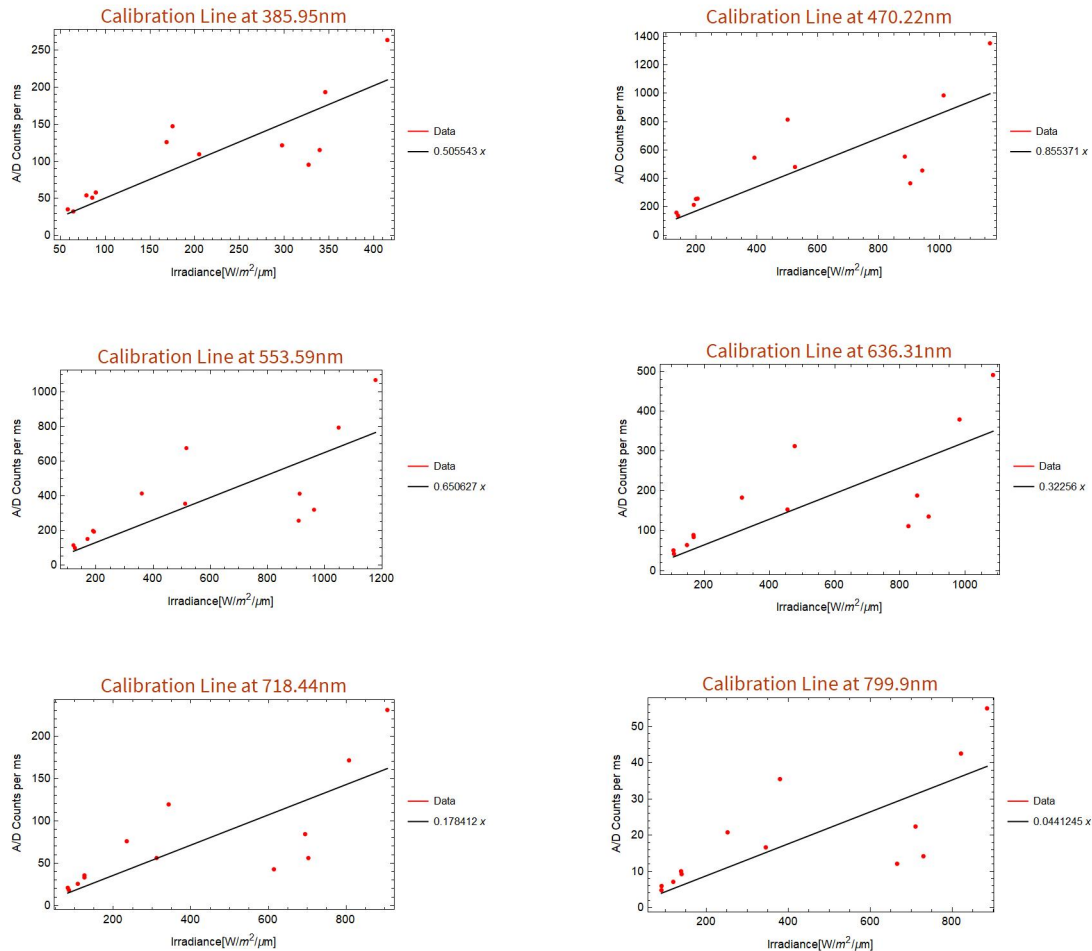


Figure 8: Example of Linear Regressions made from the gathered Spectral intensities.

The regression line is altered such that it passes $(0, 0)$. ADC is the unit of spectrometer intensities; it is a number that is converted from voltage generated by the spectrometer when light enters its sensors into digital signals a computer can process. Before we proceed, it is important to first check the fit of the regression line. To do this the Coefficients of Correlation for each linear regression are calculated and collected into one graph:

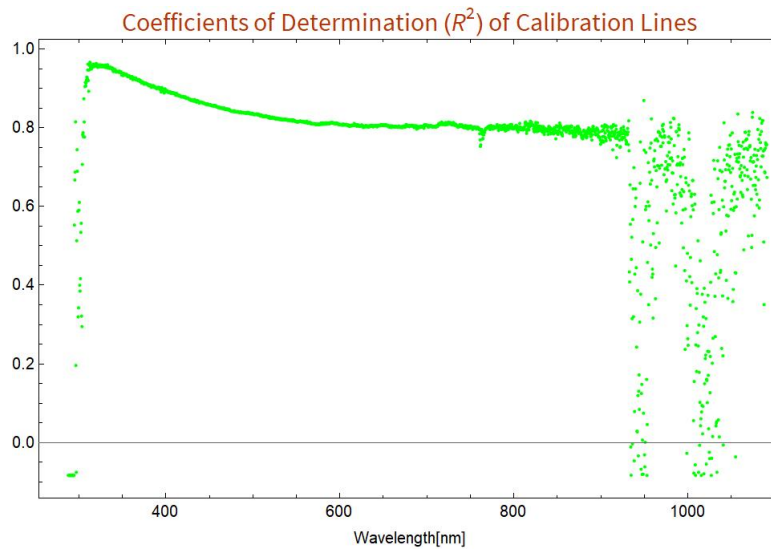


Figure 9: Coefficients of Determination of each regression line, called calibration line to represent its function. The coefficients are automatically calculated by Mathematica when making regression models.

From the graph it is shown that between $300nm$ and $900nm$ the coefficients are ≤ 0.8 out of 1, though after $800nm$ the coefficients become more noisy. Before this range we see a sharp increase from 0.05 to 0.95, and after the range it starts to oscillate between 0.8 and 0.05. This means that our regression lines are only effective between $300nm$ and $900nm$. This is due to the insensitivity of the spectrometer at wavelengths close to its boundaries. This can be seen in an example below.

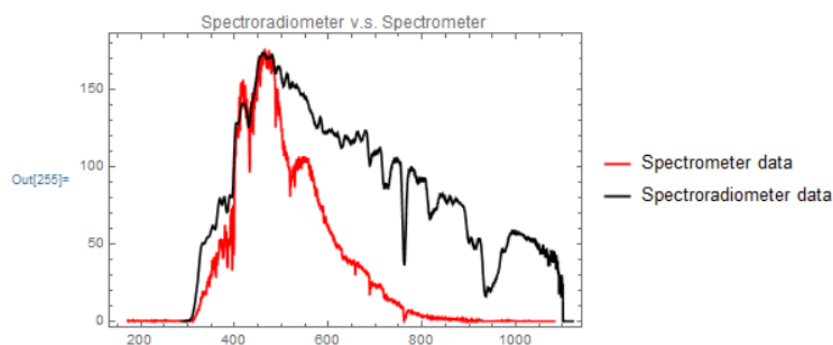


Figure 10: A comparison of spectrometer and spectroradiometer spectra. The x-axis contains the wavelengths.

It is then necessary to fix the converted spectral intensities with the transmissivity

of the diffuser. As it is not perfectly translucent, it absorbs some of the light that goes through it. This is provided by its seller Thor Labs¹⁰. The diffuser's transmissivity varies, depending on the wavelength of the spectra, and resembles a logarithmic function.

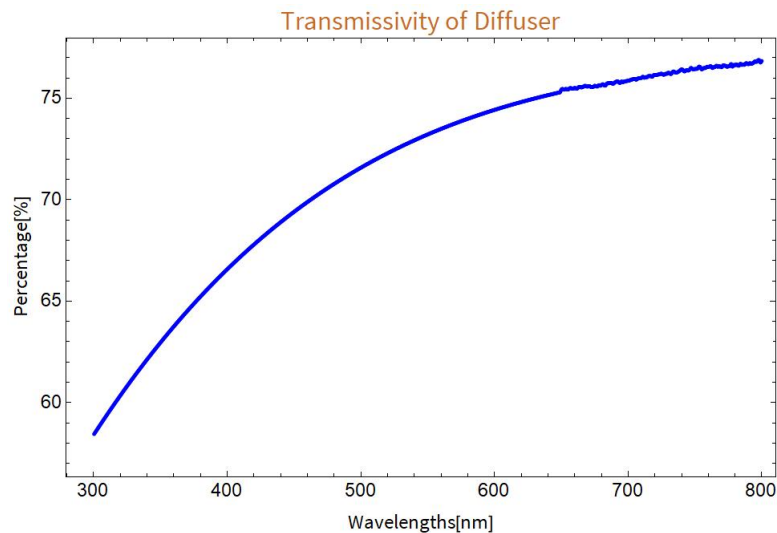


Figure 11: Percentage of light transmissivity of the diffuser depending on wavelength. Data provided by ThorLabs¹⁰, plot generated via Mathematica.

As this transmissivity contribution is presumably fixed by the previous intensity calibration, there is no need for further corrections. This is because the calibration curve will convert the raw data, which includes the diffuser contributions, into the corresponding irradiance values regardless.

Integration Time Check

One of the prime assumptions when using a spectrometer is that there is a linear relation between the integration time i.e. the time interval that the spectrometer allows light to enter before concluding a measurement, and the spectral intensity it measures. This we used when dividing the spectral intensities of each spectrometer spectrum with their integration times during the intensity calibration. Due to the age of the spectrometer it is therefore essential to check whether its original in-built calibration is still accurate, as

the spectrometer is insensitive when close to its wavelength boundaries seen in figure 10. This is tested by measuring spectra with integration times $40ms$, $80ms$ and $120ms$ in rapid succession. The values are then divided with each other. If the assumption holds then we should see the ratios to be roughly equal to divisions between the integration times of spectra. The test is done in daylight under cloudless conditions to ensure little atmospheric interventions, whereby the subsequent $40ms$, $80ms$ and $120ms$ duration light measurements are assumed to have equivalent lighting conditions.

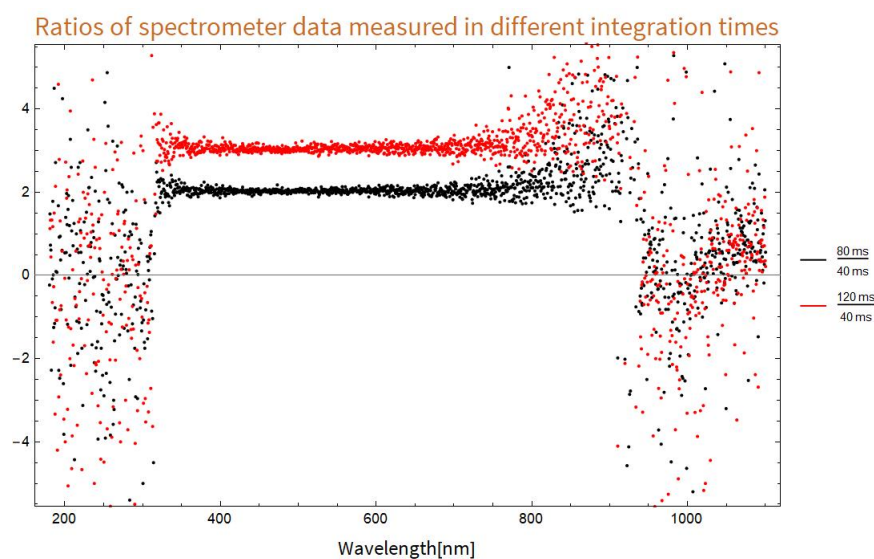


Figure 12: Ratios of spectra with different integration times. Black data points represent spectral intensities measured with $80ms$ divided by spectrum measured with $40ms$. They are supposed to coalesce around the line $x = \frac{80}{40} = 2$. Red data points represent spectral intensities measured with $120ms$ divided by spectrum measured with $40ms$. They are supposed to coalesce around the line $x = \frac{120}{40} = 3$. Not all data points are included in the plot, as some have their y-values $> 10^3$. Plot generated via Mathematica.

From the graph one can see that our assumption only holds within a certain wavelength range. For both divisions the ratios only stayed at the expected values between wavelengths of $300nm$ and $800nm$. Before this range the data points scatter around without much reason. After this range we see a gradual increase in noise in a trumpet-like fashion, until $900nm$ where the coordinates also scatter around. A lot of the points

outside the range are negative due to the in-built outdated calibration, indicating large presence of negative intensities measured by the spectrometer. This means that in these regions there is high relative noise. Thus our calibrations are only really effective between the range $300nm \leq x \leq 800nm$.

Data Processing

To compare the measured spectra with those predicted by the LAD's NN, data processing is required. The goal is to produce a calibrated spectrometer spectrum and a predicted spectrum measured at the same time that can be compared with each other. To do this we first export all the spectrometer data, hereby referred to as measured, into .csv files so both the programming software and a person can read it easily. All measured data sets of the same day will be exported into one .csv file for simple storage; there are around $(13 \text{ days}) \cdot (12 \text{ directions}) \cdot 24 = 156$ data sets in total for all directions, with five in each set. However, as three days there where partial rain not all sides are measured, resulting in $156 - 24 = 132$ datasets in total. In addition I pulled out all the LAD data within the relevant months, February to April, and stored them in a single folder for manipulation. The entire procedure is done on Mathematica, a programming software that allows efficient mathematical manipulations and representation. Due to the complexity of the operations to be done they are separated into five different codes. They are included in the appendix for reference.

The first code prepares the measured spectra for more complex manipulations and conducts the wavelength calibration. It first finds all the measurements and divides them by their respective integration times to make the former independent from the latter. Then it finds all the measurements made in the same direction of a certain time frame and averages them. This removes random errors that may exist in individual measurements. After that the wavelength calibration is applied. The resulting spectra are grouped by measurement dates and exported. Each spectrum is named by the day it is measured, with the first row encoding the direction and the time they were measured. By now the

spectra have been cleaned up and the data can be prepared for intensity calibration.

The second and third code interpolate the measured spectra and calibrates their intensities. The second code interpolates all the spectra. This is then used to get spectral values of specific wavelengths. This is for two reasons. The first is that the intensity calibration is done under spectroradiometer wavelength values. The second is that the predicted spectra from the LAD and Neural Network also uses the spectroradiometer's wavelength values, the NN being trained using the latter's outputs. After this operation the converted measured spectra are then exported as before, and inputted into the third code. The third code applies the set of calibration lines, obtained during the Intensity Calibration, on the measured spectra, converting the intensities into irradiance values $W \cdot m^{-2} \cdot \mu m^{-1}$. By this point the measured spectra are ready for comparison with output of NN. Now we move on to LAD data.

The fourth code finds the right LAD data using information from the measured spectra and inputs them into the Neural Network. It does this by reading the name of each measurement spectrum, the time at which its was measured and the direction. Then it inputs these instructions to find the right LAD data file and from it the right data entry in the right direction. The data entry is then inserted into the NN, outputting a series of irradiance values. This is then paired with the spectroradiometer wavelength values to make the predicted spectra. The predicted Spectra are then exported in a similar fashion as the measured spectra, grouped under measurement days.

The fifth code plots the predicted and the calibrated measured spectra together. This is separated to allow for fine tuning of graphical representation. This way, one can get an intuitive idea of the differences between the two spectra by looking at them. The plots have a limited x range of $300nm \leq x \leq 800nm$, since the measured spectra are only meaningful within the regions where integration time is linearly proportional with measured intensities. This concludes the basic data processing.

Further Analyses Methods

We chose four concepts to characterise the predicted Spectrum: the Total Intensities, the Noises, Over/Underestimation and the Percentage Errors. These four allows one to characterize the predicted spectra with singular values per measurement set. With these numbers it is possible to characterize the performance of the Neural Network for different light conditions than it has been trained for. The Total Intensity calculates the total intensities of the measured and predicted spectra.

The Noise is characterized by two values: the average of differences of neighboring irradiance values from each predicted spectrum, and the average of differences of neighboring irradiance values from each measured spectrum. The first shows the average distance between two irradiance values, illustrating the noisiness of the predicted result. The second shows the average of differences of neighboring irradiance values from each measured spectrum. From this the noise of the two can be compared.

The Over/Underestimation calculates the amount of irradiance values the predicted spectra has more or less than the measured spectra. It is characterized by two values. The first is the quotient of the LAD's absolute intensity with spectrometer's absolute intensity. It shows the proportional accuracy of NN predictions. The ideal value is 1, where both are the same. The second is the difference of the two absolute intensities. It shows the numerical accuracy of NN predictions. The ideal case here is when the value is 0. This is done since it is possible to have a proportionally low under/overestimation but a large difference due to large spectral values, and vice versa for low spectra values.

The Percentage Error is characterised by the standard deviation of the normalized differences between the predicted irradiance values and measured. This is to find the similarity, or lack of, in the shape of the predicted and measured spectra. The normalized differences between the two spectra indicates how different irradiance values are at each wavelength. By finding its standard deviation, we get a value that indicates how much do the differences on average deviate from the average difference. Should it be low, it means that the differences are similar in value and therefore the two spectra would have a similar shape. On the contrary situation, the differences' values varies thus resulting in two spec-

tra of different shapes. To help understand the value the mean difference between each NN spectrum and measured spectrum is also included. This shows the values that the normalized differences i.e. percentage error are deviating from or to, which would assist understanding the metric.

We recall a predicted Spectrum to be a series of irradiance values produced by the Neural Network through input from LAD values. These are assigned to unique wavelength values sourced originally from the spectroradiometer spectra that the Network is trained upon: $S_{predicted} : Wavelength_i \rightarrow E_{eP_i}$. The measured spectrum can be defined in a similar fashion: $S_{Reference} : Wavelength_i \rightarrow E_{eR_i}$. Both sets of data share the same number of value sets and same wavelengths. Due to the effective range of the spectrometer we limit the values from around $300nm$ to $800nm$. The closest wavelength values from the spectroradiometer to the established range is $300.38nm$ and $799.9nm$, which are the 30th and 1224th wavelength values respectively, sorted by ascending order. Therefore we define i to have the range $30 \leq i \leq 1224$. Using this we can express the 3 concepts in mathematical terms:

$$\begin{aligned}
 \text{Total Intensities: } & \begin{cases} \sum_{i=30}^{1224} E_{eR_i} \\ \sum_{i=30}^{1224} E_{eP_i} \end{cases} \\
 \text{Noise: } & \begin{cases} \frac{1}{N} \sum_{i=30}^{1223} |E_{eR_{i+1}} - E_{eR_i}| \\ \frac{1}{N} \sum_{i=30}^{1223} |E_{eP_{i+1}} - E_{eP_i}| \end{cases} \\
 \text{Over/Underestimation: } & \begin{cases} \sum_{i=30}^{1224} E_{eR_i} / \sum_{i=30}^{1224} E_{eP_i} \\ \sum_{i=30}^{1224} E_{eR_i} - \sum_{i=30}^{1224} E_{eP_i} \end{cases} \\
 \text{Percentage Errors: } & \begin{cases} \frac{1}{N} \sum_{i=30}^{1224} (E_{eR_i} - E_{eP_i}) / E_{eP_i} \\ \sum_{i=30}^{1224} \sqrt{\sum_{i=30}^{1223} (x_i - \mu)^2 / N} \end{cases} \\
 \mu = & \frac{1}{N} \sum_{i=30}^{1224} (E_{eR_i} - E_{eP_i}) / E_{eP_i}, \\
 x_i = & (E_{eR_i} - E_{eP_i}) / E_{eP_i}
 \end{aligned}$$

Results

Reference Spectra v.s. Predicted Spectra

Here a portion of the comparison plots for all directions is shown. The entire set of data will be shown in the appendix. Figure 13 shows an example of spectral plots of directions A to F, and the following figure shows the same thing but for directions G to L.

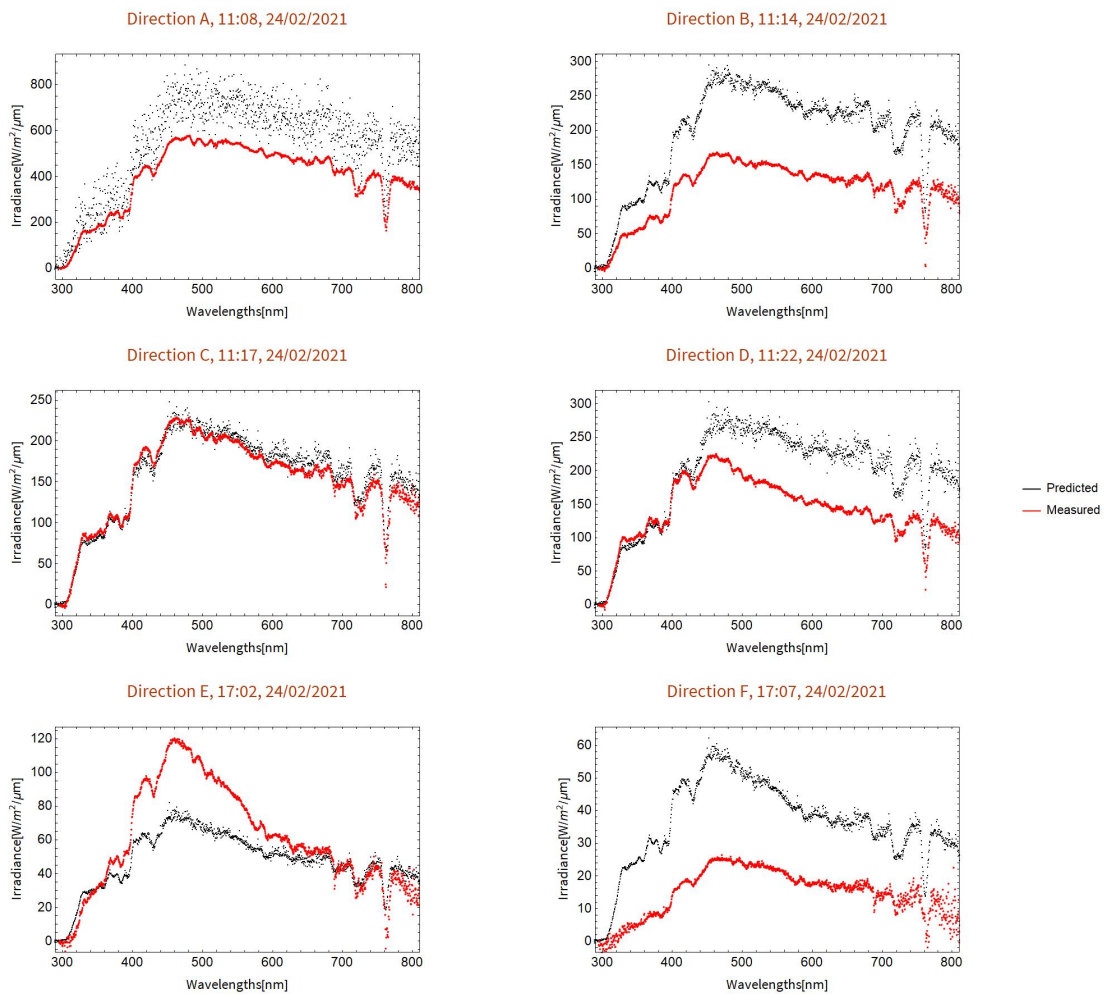


Figure 13: A set of Spectral data comparison in all 12 directions on 2021/02/24. The graph titles tell by order, the direction, the time and the date when it is measured. As legend indicates black dots are predicted spectrum using NN and LAD data and red ones are measured with the spectrometer.

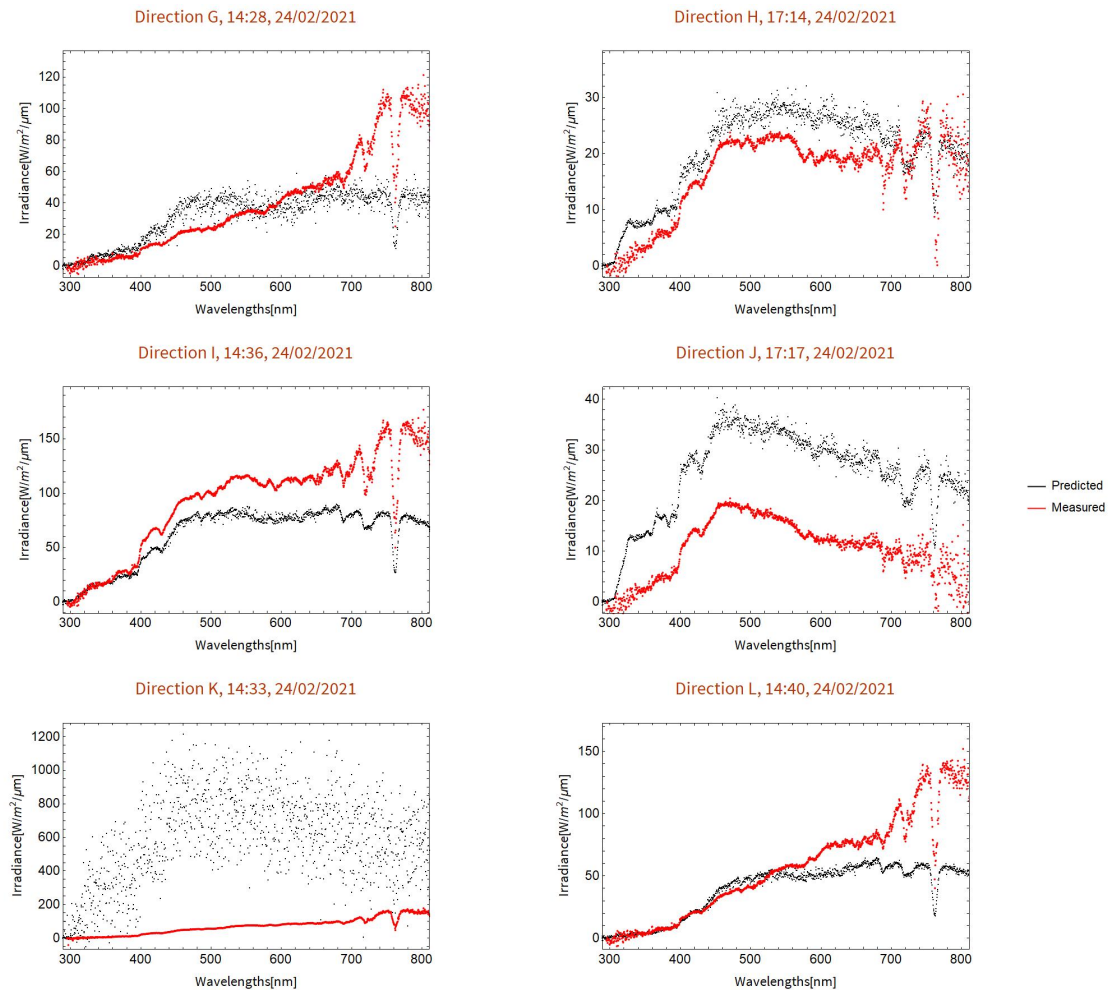


Figure 14: A set of Spectral data comparison in all 12 directions on 2021/02/24. The graph titles tells by order, the direction, the time and the date when it is measured. As legend indicates black dots are predicted spectrum using NN and LAD data and red ones are measured with the spectrometer.

These figures shows the an example of a set of measured & predicted spectra. All 12 directions of the dodecahedron are shown here, measured in each face once at different times but on the same day. Certain trends are visible in these graphs. First and foremost, at high intensities, i.e. peak intensity $\leq 800 W \cdot m^{-2} \cdot \mu m^{-1}$, we see that the predicted spectra have a high amount of noise. Prime examples being Direction A and K. Secondly, directions G, K and L, and to some extent I have the strong irradiance differences between the measured and predicted spectra. Out of these directions K is especially egregious, having quite a few intensities that are much more intensity and of different shape than

the measured spectra, as one can also see from the graph. Direction E also has a similar issue (in this case around $500nm$).

Further Analyses

Total Intensities

In figures 15 and 16 the Total intensities, $\sum_{i=30}^{1224} E_{eP_i}$ for Predicted and $\sum_{i=30}^{1224} E_{eR_i}$ for reference spectra, are calculated and plotted.

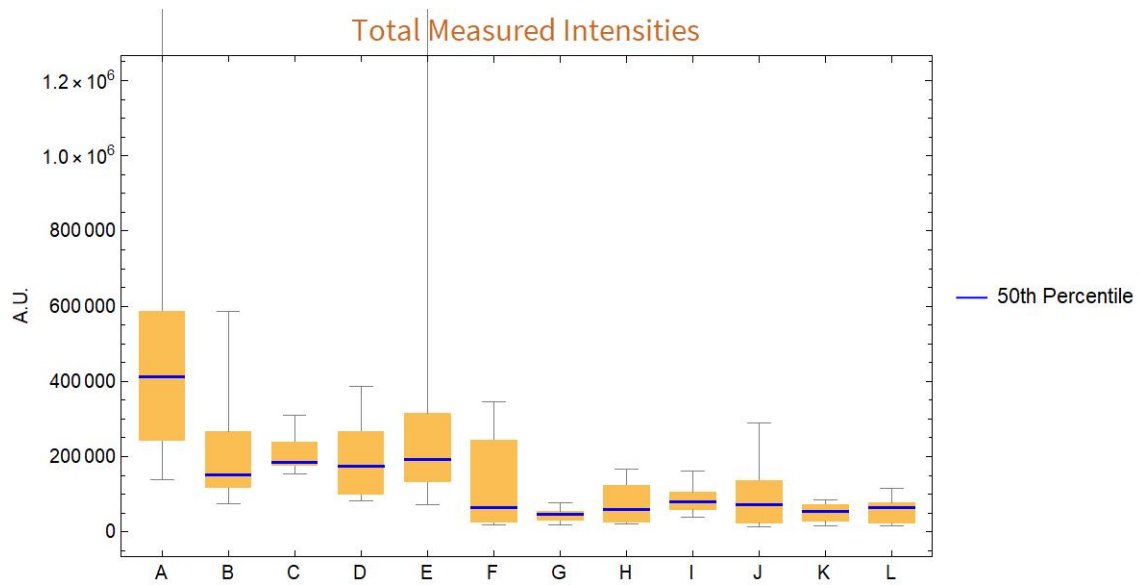


Figure 15: Total Intensities of reference spectra. For effective comparison the points $\approx 1.48 \cdot 10^6$ of direction A and $\approx 3.29 \cdot 10^6$ of direction E are not shown in here.

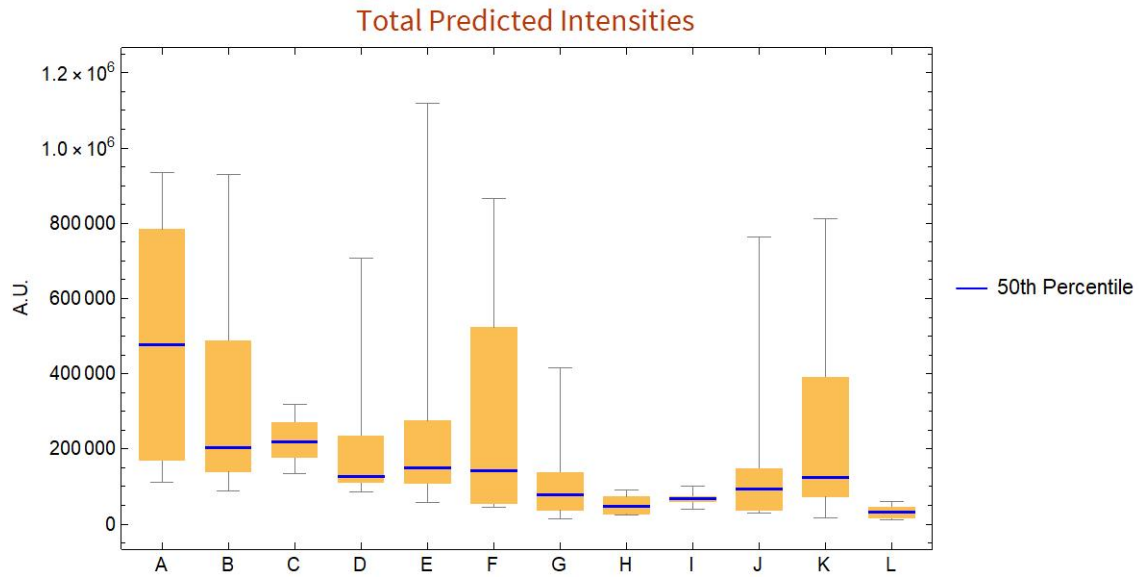


Figure 16: Total Intensities of predicted spectra.

Judging from figures 15 and 16 one can identify a few trends visually. First for directions that faces down (G, H, I, J, K & L) the measured and predicted spectral intensities are in general lower compared to directions that faces up (A, B, C, D, E & F). For Predicted intensities there is the exception of direction K, with its total intensity in the same order as direction A, B and F. This large difference between the measured and predicted spectra in direction K shows a low level of NN reliability in this direction. This would be expected to be reflected in other upcoming analyses where both spectra are used for calculations. However, this may not be the case for noise calculations since for these facing down directions the low intensities mean that they would probably have less noise than their sky-facing counterparts.

Noise

For Noise we measure the spectral noise of the Measured and Predicted Spectra with the equations $\frac{1}{N} \sum_{i=30}^{1223} |E_e R_{i+1} - E_e R_i|$ and $\frac{1}{N} \sum_{i=30}^{1223} |E_e P_{i+1} - E_e P_i|$.

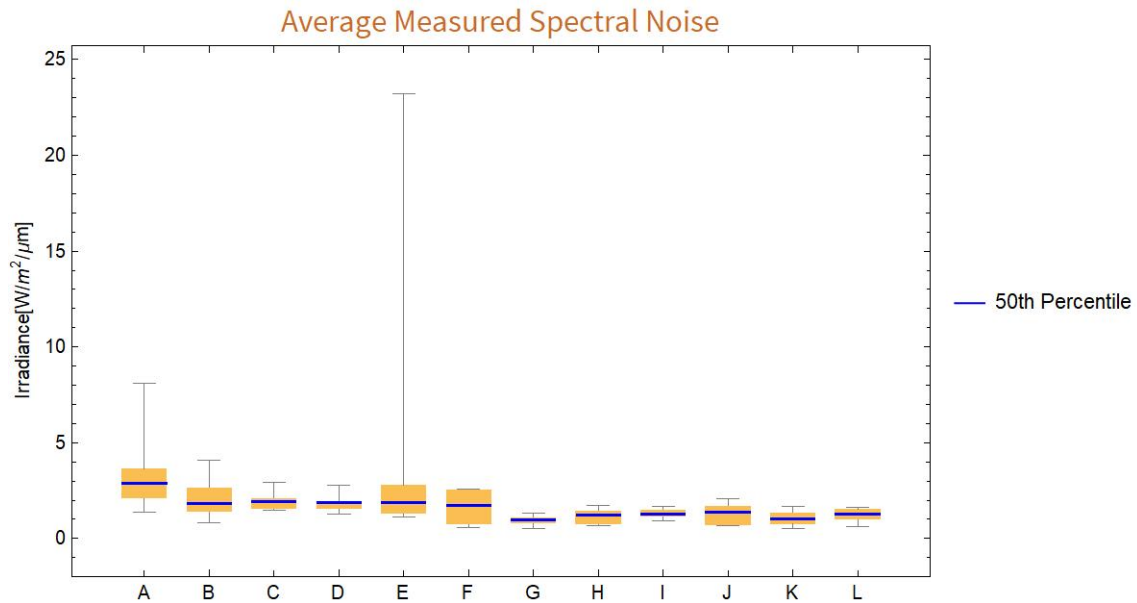


Figure 17: Mean of neighboring irradiance values' differences of reference spectra.

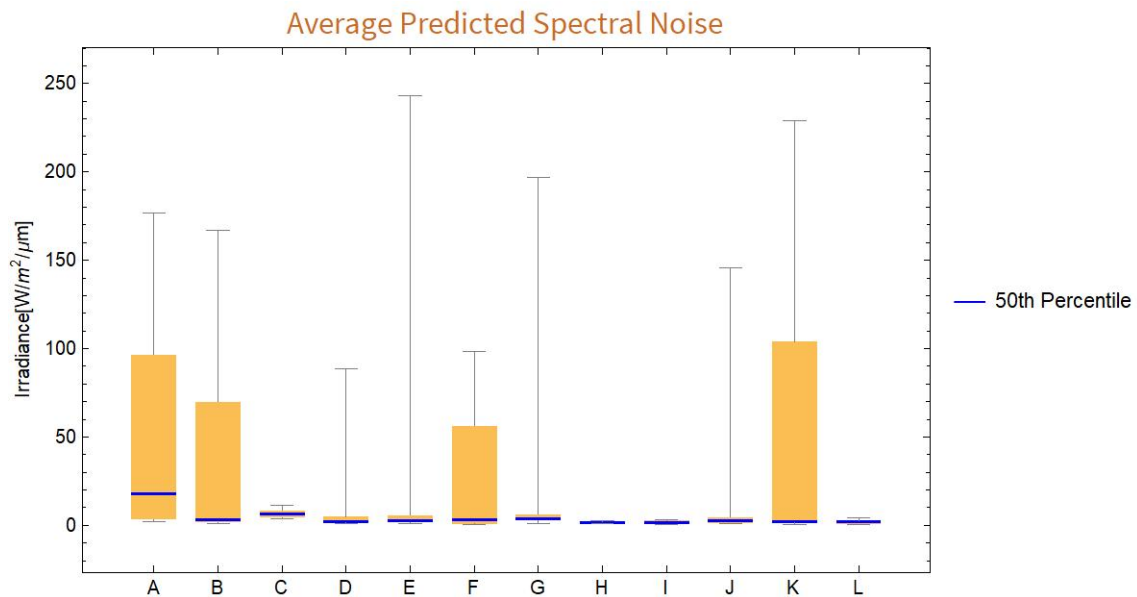


Figure 18: Mean of neighboring irradiance values' differences of predicted spectra.

From the figures it is clear that directions A, B, F and K have much more noise than the other directions for predicted spectra, having their 75th percentile $\geq 50 W \cdot m^{-2} \cdot \mu m^{-1}$. While the others have theirs $\leq 10 W \cdot m^{-2} \cdot \mu m^{-1}$. Curiously, direction H has smallest noise, a trait one sees also in the previous section. This is very likely due to the fact that direction H for both measurement tools faces a large expanse of greenspace, tilted

slightly down. While there are trees in its way they are very thin and thus are of little consequence. The buildings in the background are not reflective. This may have allowed direction H of the two tools to be immersed in similar diffuse light, thus resulting in the good predictions. Directions A, B and F are as expected since they are directions expected to measure high intensities, demonstrated by figure 18.

We find that a majority of directions has relatively little noise. Of these directions (C, D, E, G, H, I, J and L) only have six value in the order of 1 and three in the order of 2 out of 86 data points; proportionally $\approx 10\%$ of the total amount of data have values higher than 10. Comparatively, the other directions (A, B, F and L) have eight values in the order of 1 and seven values in the order of 2 out of 44 data points; proportionally $\approx 34\%$ in total are higher than 10. Considering that the NN is trained in direction A, the fact that 8 directions performed much better than it means that the NN is capable of producing relatively noiseless data when compared to direction A. Of these ones we see that directions closer to the sky and direct Sunlight e.g. C, D and E are usually, with the exception of direction K, more noisy than the lower directions e.g. I, J & L. This is very likely again caused by the NN's aforementioned weakness of making noisy predictions at high irradiance values. For direction K the large noise could be due to reflections from a large body of water and buildings with large windows in its field of view.

Over/Underestimation

The purposes of this section is to tell by how much or less the NN predicts in terms of total intensity. This is done by computing two different mathematical operations for all directions: $\sum_{i=30}^{1224} E_e R_i / \sum_{i=30}^{1224} E_e P_i$ and $\sum_{i=30}^{1224} E_e R_i - \sum_{i=30}^{1224} E_e P_i$. The former finds the factor of over/underestimation and the latter calculates the difference between the two.

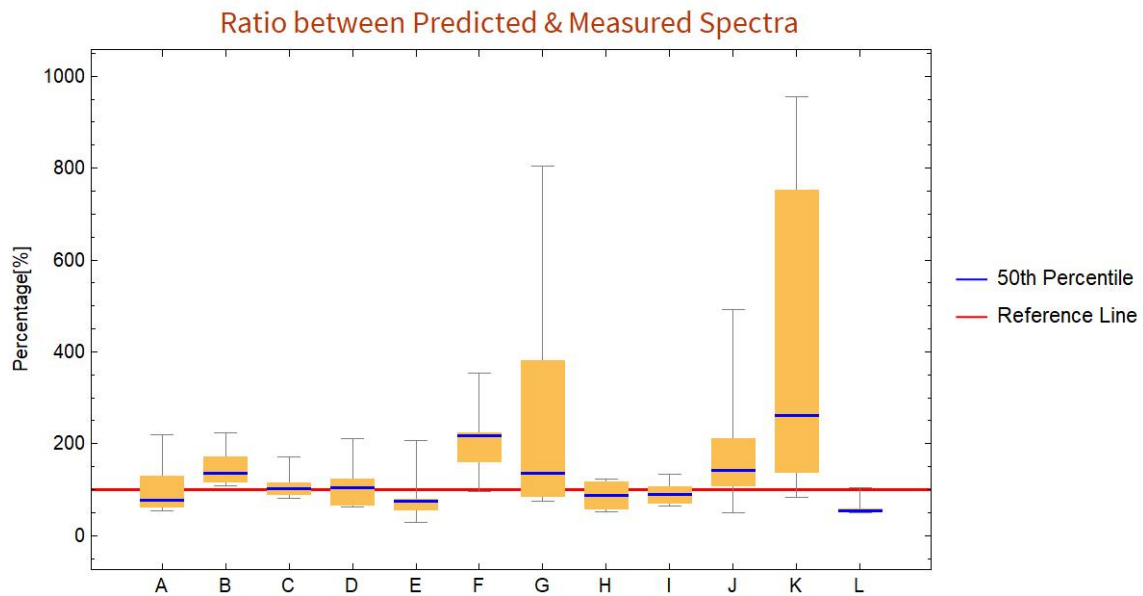


Figure 19: Measures of proportional inaccuracies of prediction of absolute intensities in each direction. Factors have all been multiplied by 100 to show percentages.

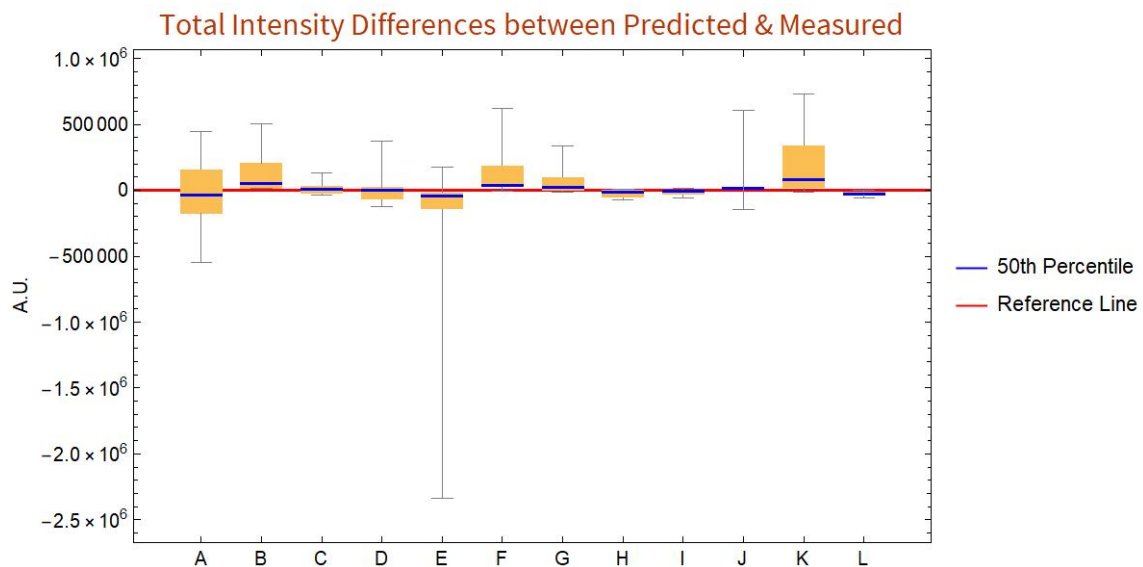


Figure 20: Measures of numerical inaccuracies of prediction of absolute intensities in each direction.

We see in figure 19 for directions B, C, F, G, J & K the 50th percentile is above the reference line and for A, E, H, I, L it is below. This shows that the NN overestimates the irradiance intensity for the former and underestimates for the latter. For direction D the irradiance of the predicted and measured spectra match well as the 50th percentile lies

approximately on the line, seen in figure 19. For figure 20, direction D's 50th percentile is almost on the reference line. Thus we can say that, excluding outliers, the Neural Network is not necessarily biased towards overestimation nor underestimation. This is supported by the fact that of all the spectra 59 of them overestimates (percentage wise $\frac{59}{132} \approx 45\%$) and 73 underestimates ($\frac{73}{132} \approx 55\%$). While there is $\approx 10\%$ more overestimation, it is not a large number. However, it is arguable that when it overestimates proportionally, it tends to predict more than compared to when it underestimates proportionally. This demonstrated by the fact that should one add up all the proportional over/underestimations and average them, one gets on average overestimation of $\approx 161\%$ if it overestimates and average underestimation of $\approx 33\%$ if it underestimates. This is not reflected in the differences, where on average it overestimates by $\approx 1.36 \cdot 10^5 W \cdot m^{-2} \cdot \mu m^{-1}$ but underestimates by $\approx 1.44 \cdot 10^5 W \cdot m^{-2} \cdot \mu m^{-1}$. This is probably due to the minimum in direction E where numerically it is a massive negative outlier but proportionally limited. For specific directions, the strong under-/over-estimations in directions A, B and F are expected. Due to the Network's weakness of producing consistently noisy plots at high intensity outputs, the three directions, exposed to direct Sunlight at noon (A, F) and early afternoon (B), would produce inaccurate predictions. Directions G and K also have a wide range of inaccuracies surprisingly. It is possibly due to the fact that for the spectrometer setup these directions mostly face the solar field box, which is dark gray. The LAD's directions G and K not facing the box as much as the Spectrometer's; they face more of the grassy ground and are further from the box than the spectrometer dodecahedron. It could cause consistently low absolute intensities relative to the predicted spectrum, resulting in a systematic overestimation.

Percentage Errors & its Standard Deviations

For the Shape we consider the average percentage errors and the Standard Deviations of them in all directions. They are calculated by $\frac{1}{N} \sum_{i=30}^{1224} |E_e R_i - E_e P_i| / E_e P_i$ and $\sum_{i=30}^{1224} \sqrt{\sum_{i=30}^{1223} (x_i - \mu)^2 / N}$ ($\mu = \frac{1}{N} \sum_{i=30}^{1224} (E_e R_i - E_e P_i) / E_e P_i$, $x_i = (E_e R_i - E_e P_i) / E_e P_i$).

The latter estimates the average deviancy of the predicted spectral shape from its Measured counterpart. The former is the average percentage error used by the latter to calculate the level of deviation; the standards for us to find the level of deviation from the Measured Spectra.

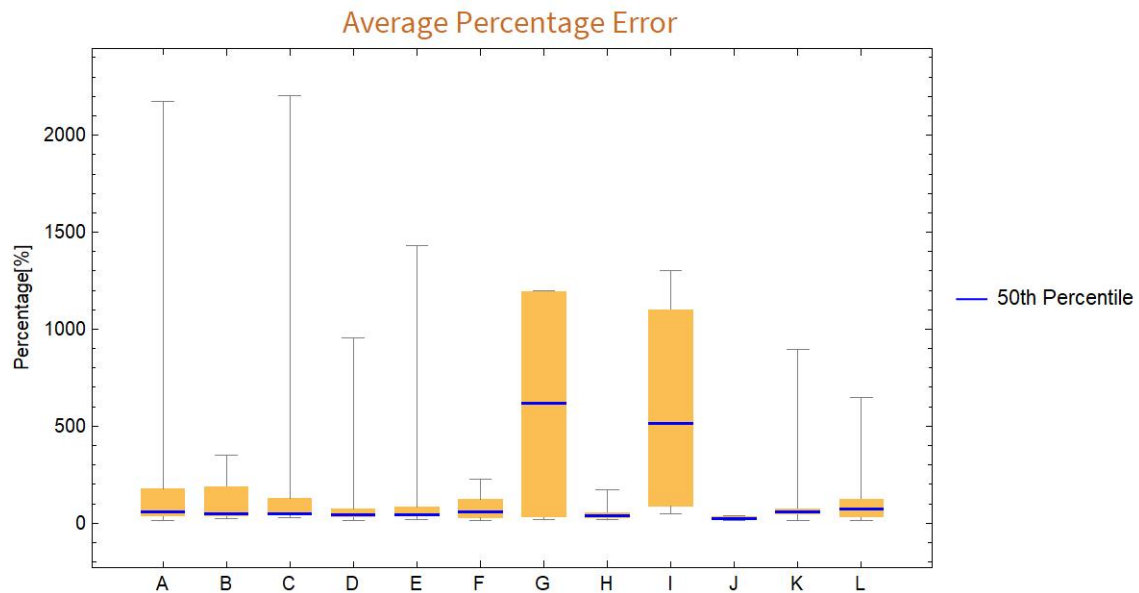


Figure 21: Mean of Absolute values of Percentage Errors of predicted spectra compared to reference.

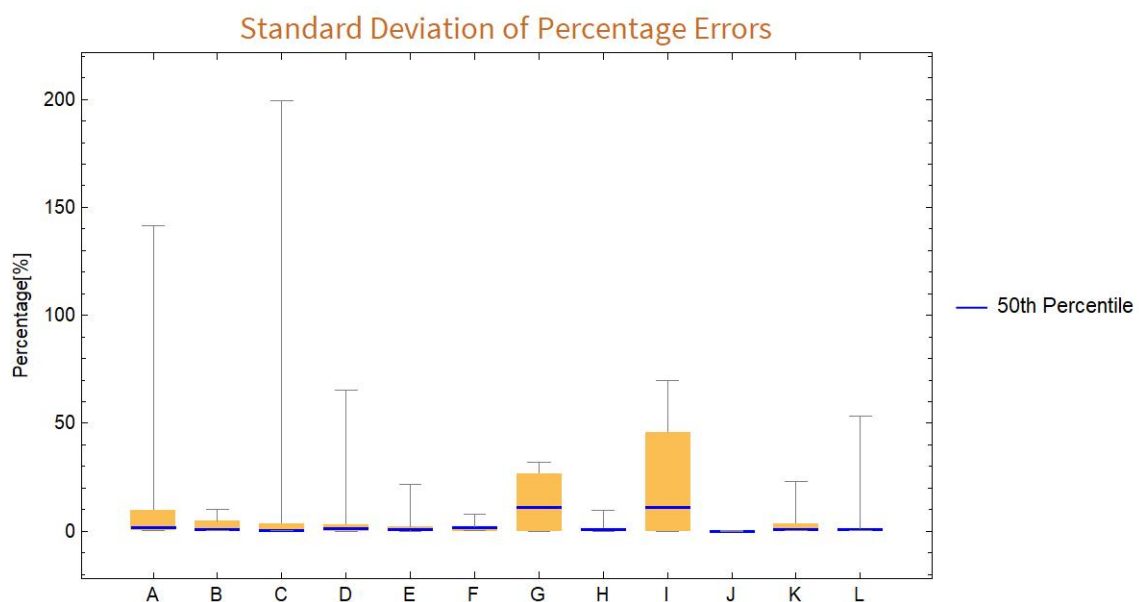


Figure 22: Standard Deviations of Percentage Errors.

The figures shows that direction G, I and K have a much wider error than the other

9 directions. While one can seek to explain these observations with the fact that the NN is trained in direction A, with light conditions much different from the directions facing down, this does not explain why direction H and J have relatively low values, indicating a good fit of the spectral shape. Looking at the actual comparisons between measured and predicted spectra, one example being 14, it seems to be that measured spectra from direction H and J happen to have a similar shape to the NN's prediction. The causes of this is not clear. This, however, does not explain why direction I has such a large variation in shape fits. It could be due to the fact that for direction I the Spectrometer is facing partially the AMOLF building and the grassy ground, allowing a combination of diffuse light conditions that is different from the LAD's direction I field of view for background. Considering direction I spectra tend to have low intensities, it requires less background surface differences to have a noticeable shape infidelity.

The other directions are as expected. Directions A, B, F and L have a low spread in differences between predicted and measured spectra but higher than the rest. In addition to having its field of view mostly to the ground, direction L has low intensities and thus is more prone to spectrometer insensitivity. Directions C, D and E show a better fit of the shape than the former set as they mostly face the sky, thus have a consistent field of view, but do not suffer from high intensity contributions as they do not face the sun directly.

Discussions & Conclusion

The metrics for analyses sheds light on the neural network's (NN) ability to predict spectra. First we look at the under-/over-estimation. Considering large values in general can cause large numerical differences only figure 19 will be referenced. From it one can see that the ratios of a majority of spectral predictions, sides B, C, D, E, H and I, are in the same order as that of side A. This indicates a certain level of accuracy. It can then be argued that the NN's ability to predict the size of the spectral intensities shows promise. We then move on to the noise.

The NN is good at making predictions with limited spectral noise in directions not side A, compared to side A. This is seen in figure 18, where directions B, C, D, F, H, I, J and L have their average noise lower in general than that of side A's. If we discount the last 25 percentile, E and G also fit the description; their last quartile is most likely outliers and thus are not wholly representative of the predictiveness of noise in those specific directions. From this it is clear that the NN can predict spectra with a relatively high degree of competency. However, the same cannot be said for the metric of percentage errors and its goal at predicting spectral shapes.

While the effectiveness of NN's predictions in predicting spectral shapes is somewhat reasonable, the metric used is not representative of shape deviancy. The metric in question is the Standard Deviation of Percentage Errors. From it we see that all non-A sides, excluding sides G and I, have standard deviations lower than what side A has. However, the method relies on the fact that the average of percentage errors, shown in figure 22, is in the middle of the range of percentage errors. If closer to the maximum or minimum, the resulting deviation can be skewed and thus do not represent the level of shape deviancy accurately. Such is but one reason that can cause high SD values without having a large deviancy. This leads to the conclusion that the NN cannot be judged to be good at producing good spectral shapes.

The NN is not effective in differentiating spectral differences in different directions. This is supported by the fact its ability to predict spectral shape is questionable, as argued above. In addition from the simple comparisons between the predicted and reference

spectra we see that while the predicted spectra can resemble the reference measurements, it would seem that the differences between the two is just as large. If one were to make a matching game, it is unclear if one can match the right directions together. This is further supported by the limited accuracy in predicting total intensities, as while other directions' proportional accuracies are in the neighborhood of side A, in terms of absolute accuracy (comparing with the red reference line seen in figure 19) it is far from sufficiency; a considerable portion of spectral ratios straying from 100%. This problem is further compounded by the fact that the NN predictions are noisy and imprecise.

The analysed metrics from NN predictions have a lot of spectral data with large outliers and some of them have large spectral noises. The former is seen in the four metrics, where in all of them at least two sides will have its 75th to 100rd percentile larger or equal to the rest of the 75% of data. The latter can be seen with the noise and percentage error metrics. In the former as seen in figure 18 sides B, F and K have relatively considerable noise, comparable or higher than side A's noise. In the latter in figure 21 we see a high percentage error in directions G and I, much higher than 75% of the data from side A. In sides B and C we see percentage errors not as high as G and I, but close to side A consistently. Considering the two problems one can argue that the NN is not ready yet for direct measuring of diffuse light as intended.

The NN is not yet usable for deciding optimal orientation of a bi-facial solar panel. This is determined by three reasons. First, its abilities to measure the spectral shapes are questionable. Secondly, the NN qualitatively is not good at predicting spectra in general in different directions, at least not to the human eye when directly comparing them with the reference spectra. Thirdly, its predictions are noisy and have one too many outliers. One may speculate the sources of these errors. Aside from the known issue of the NN being unreliable when receiving high spectral intensities⁹, which is probably the source for errors in sides A, B and F, there are also others. Some are due to the environment it is performed under and others due to the quality of the equipment.

We first consider the environmental factors. The NN's measurements are affected by the complex background setting resulting in a myriad of diffuse light conditions. In the out-

doors environment where the experiment is performed, the surroundings are covered with a variety of substances that react with light differently. Certain backgrounds e.g. water bodies and windows even reflect large amounts of light. The terrain around is also far from even, with the AMOLF building in close proximity and the solar field underneath the LAD. This is further compounded by the fact that the LAD has a different position from spectrometer dodecahedron. This causes the above reason to have a much larger effect on measurements; if one device measures the distortion but the other doesn't then the resulting spectra would not match. This is very likely the case on side K, where we see a large discrepancy in the predicted and measured spectra, and performed badly in the metric of over-/under-estimation. The same issue can be seen on side G. This unfortunately is not easily fixable since the closer it is they affect each other's reading by making their own diffuse light conditions.

We then consider the quality of the equipment. First and foremost, the age of the spectrometer, ten years, makes it unreliable as a reference. This is true in wavelengths smaller than $\approx 300\text{nm}$ and larger than $\approx 800\text{nm}$, where the assumed positive relation between the integration time and spectral intensities are not kept. While this is excluded from the analyses, effectively intensities from 45% $(1 - \frac{800\text{nm}-300\text{nm}}{1100\text{nm}-180\text{nm}})$ of wavelengths are lost. This means that the results obtained previously is not enough to conclude the usefulness of the NN and the LAD for bi-facial cells even if the aforementioned errors are solved. Then there is the fact that the angle of acceptance of the spectrometer setup ($\frac{\pi}{6}$ radians) is different from spectroradiometer that is used to calibrate it ($\frac{\pi}{2}$ radians) and the LAD ($\frac{\pi}{2}$ radians). In addition the dodecahedron of the spectrometer setup may have slight differences in angle orientation compared to LAD, resulting in consistent variations of the amount of diffuse light measured as it is highly direction-dependent.

More conclusive results can be obtained if similar tests are done under a uniform background e.g. a desert with well-calibrated spectrometer and diffuser with the right acceptance angle. Alternatively, using the spectroradiometer instead to measure would work as it is calibrated, but since it is fixed onto the solar field that's not very tractable.

Appendix

Complete set of Reference Spectra v.s. Predicted Spectra

For the following graphs, the titles tells by order, the direction, the time and the date when it is measured. As legend indicates black dots are predicted spectra and red ones are measured.

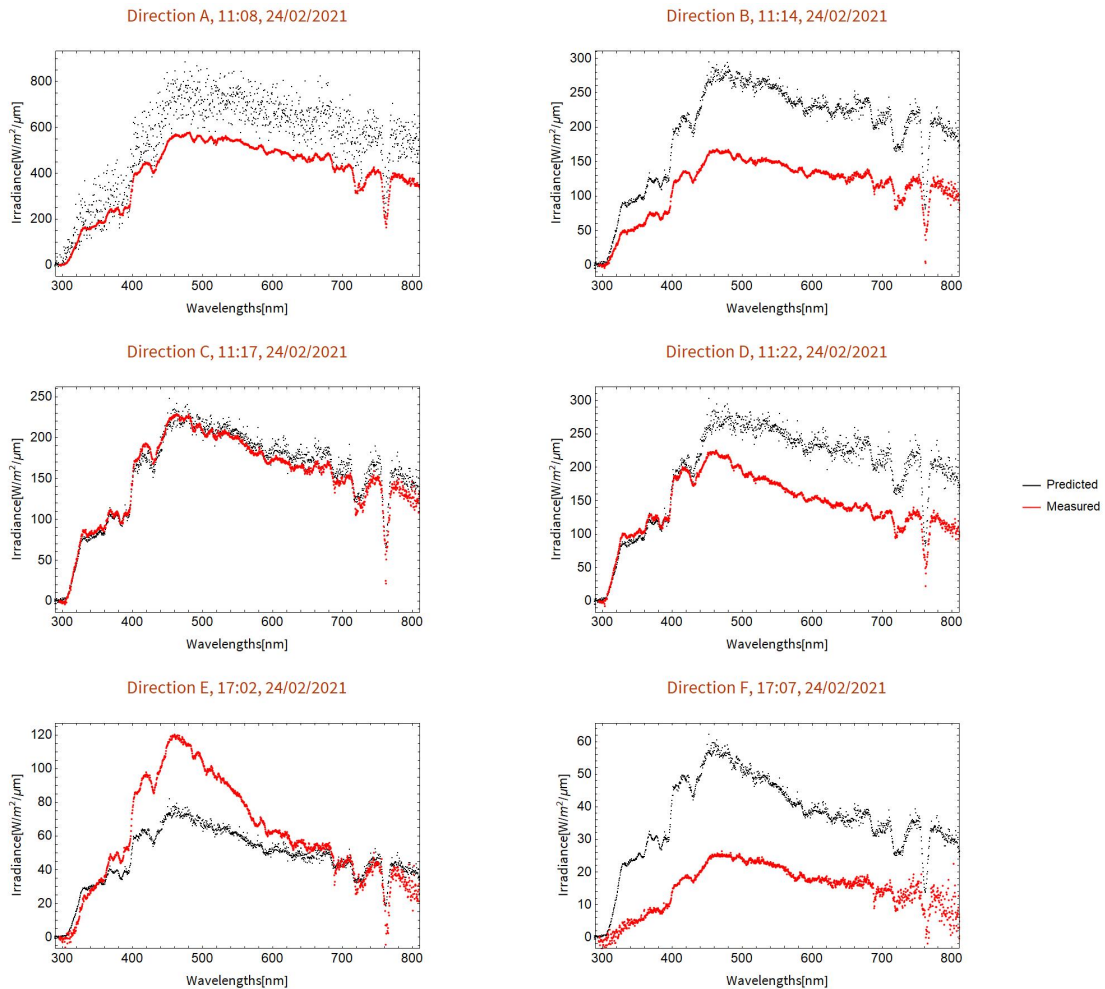


Figure 23: A set of Spectral data comparison in directions A to F on 2021/02/24.

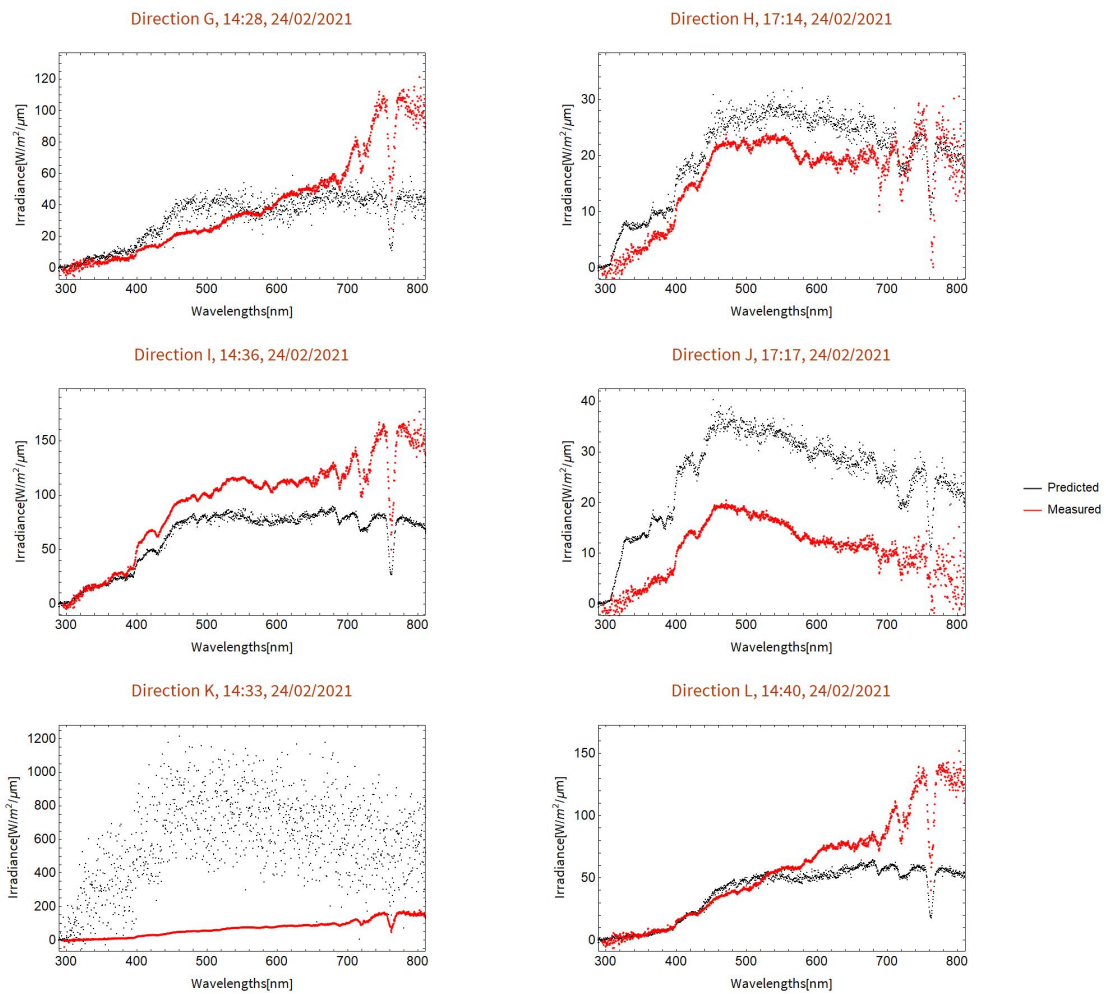


Figure 24: A set of Spectral data comparison in directions G to L on 2021/02/24.

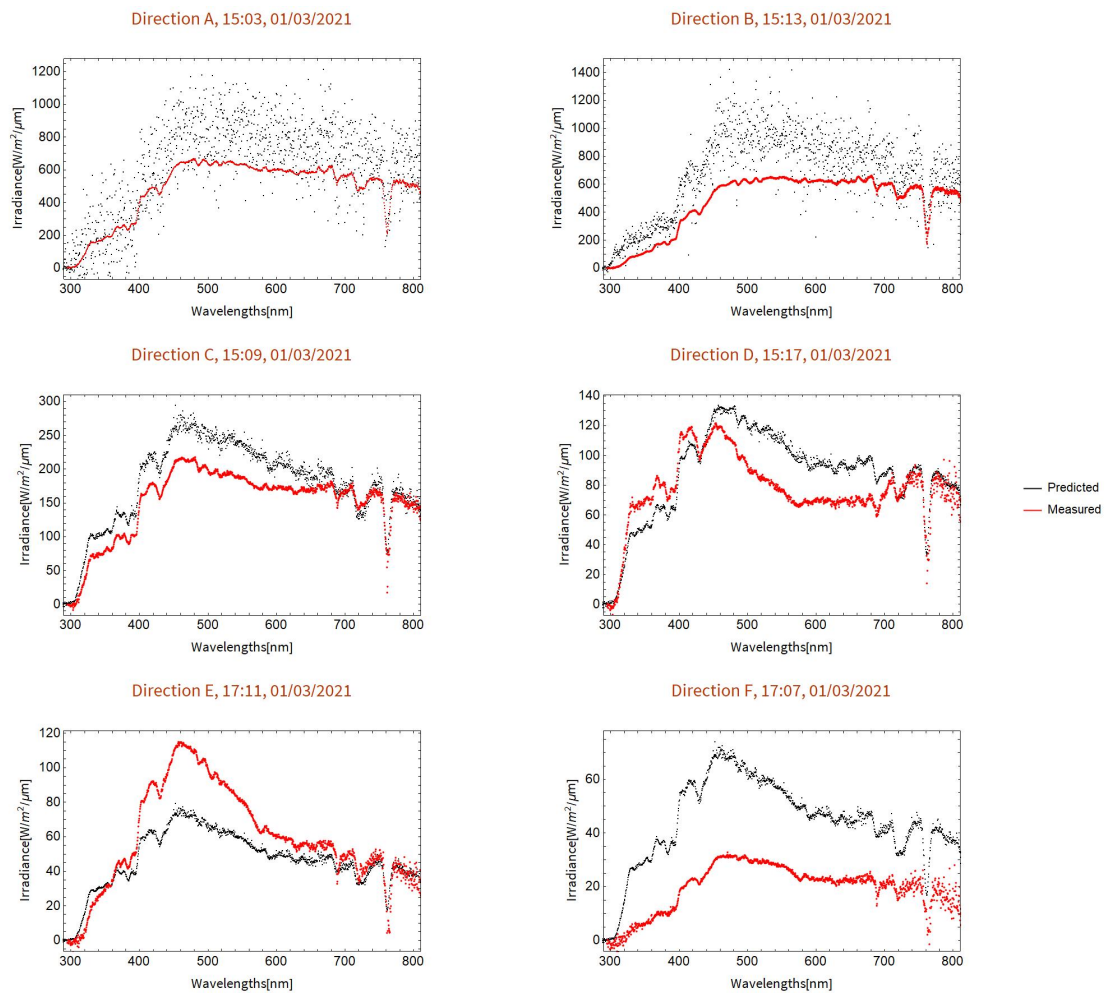


Figure 25: A set of Spectral data comparison in directions A to F on 2021/03/01.



Figure 26: A set of Spectral data comparison in directions G to L on 2021/03/01.

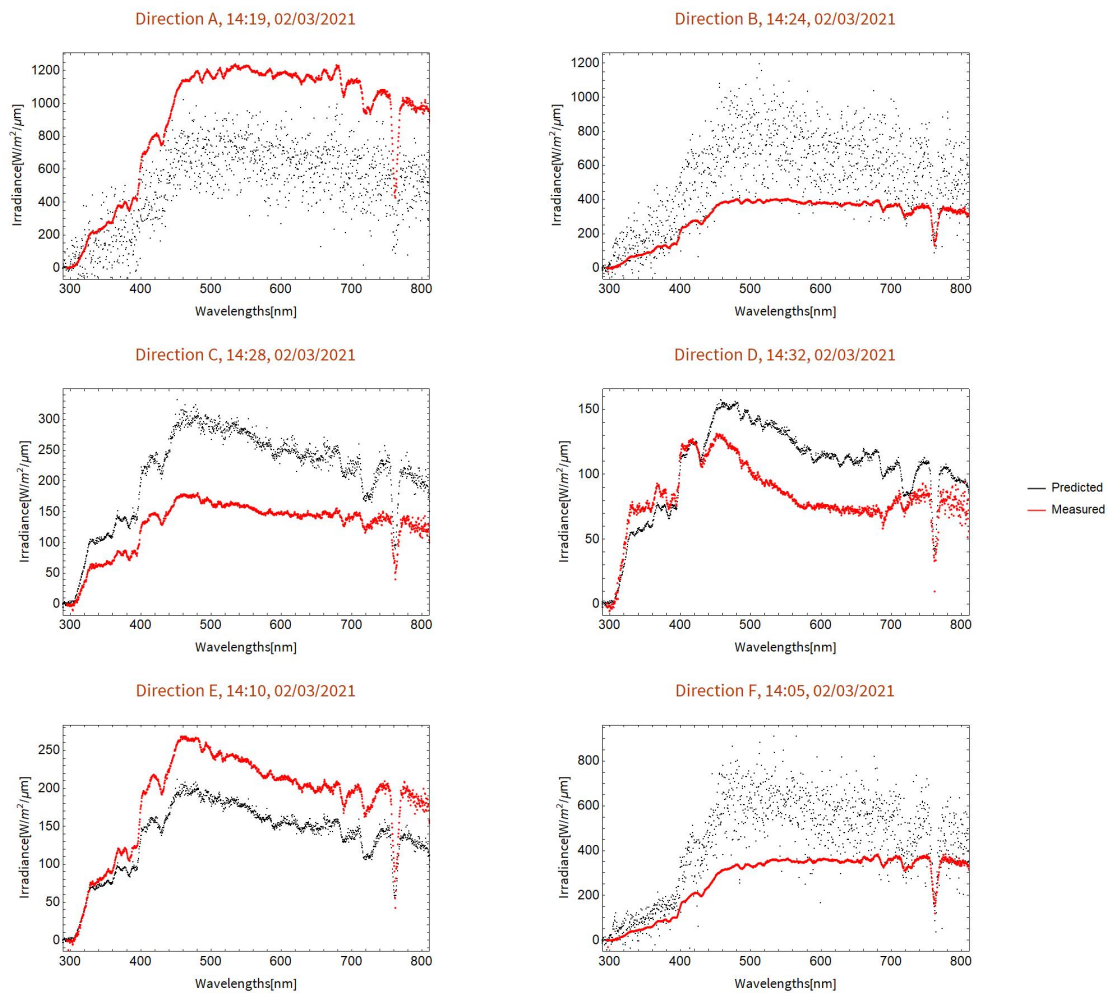


Figure 27: A set of Spectral data comparison in directions A to F on 2021/03/02.

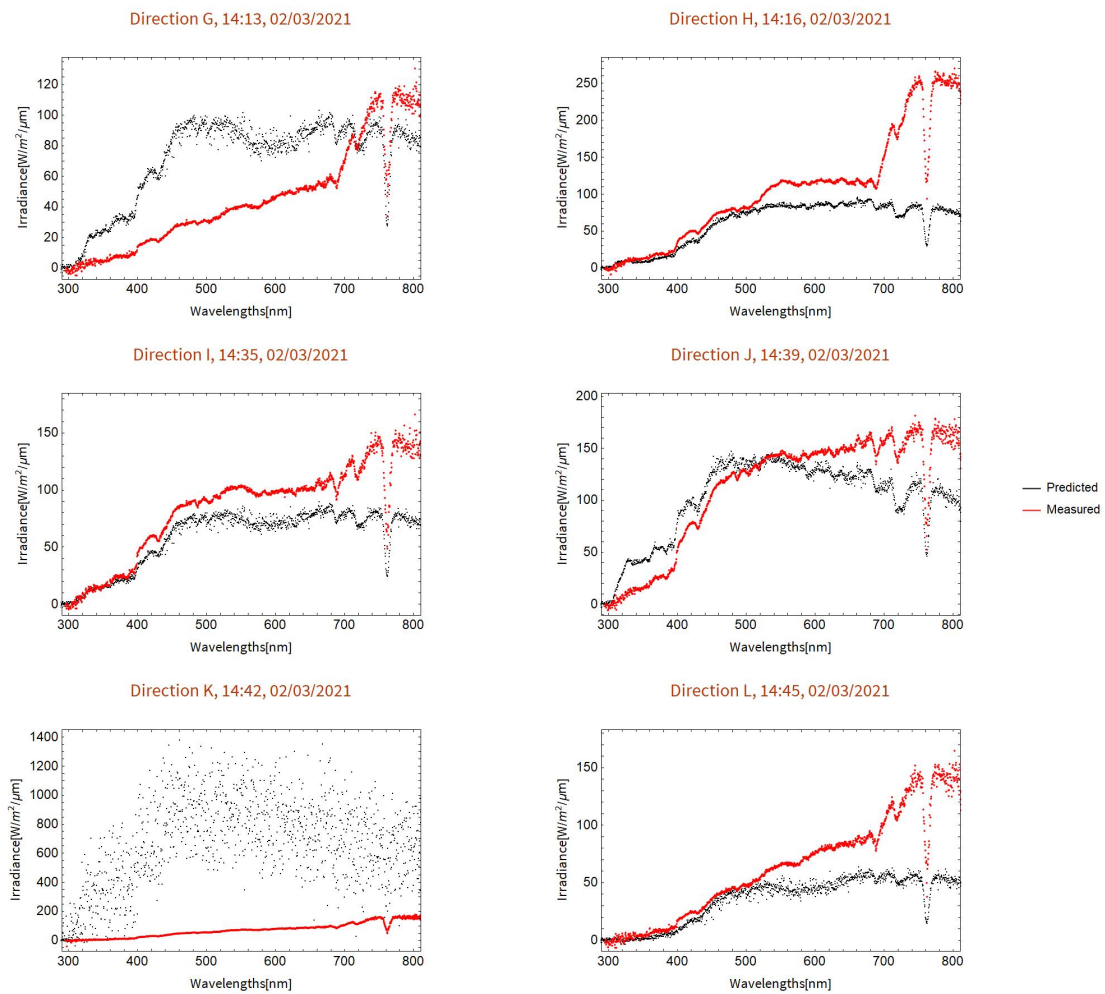


Figure 28: A set of Spectral data comparison in directions G to L on 2021/03/02.

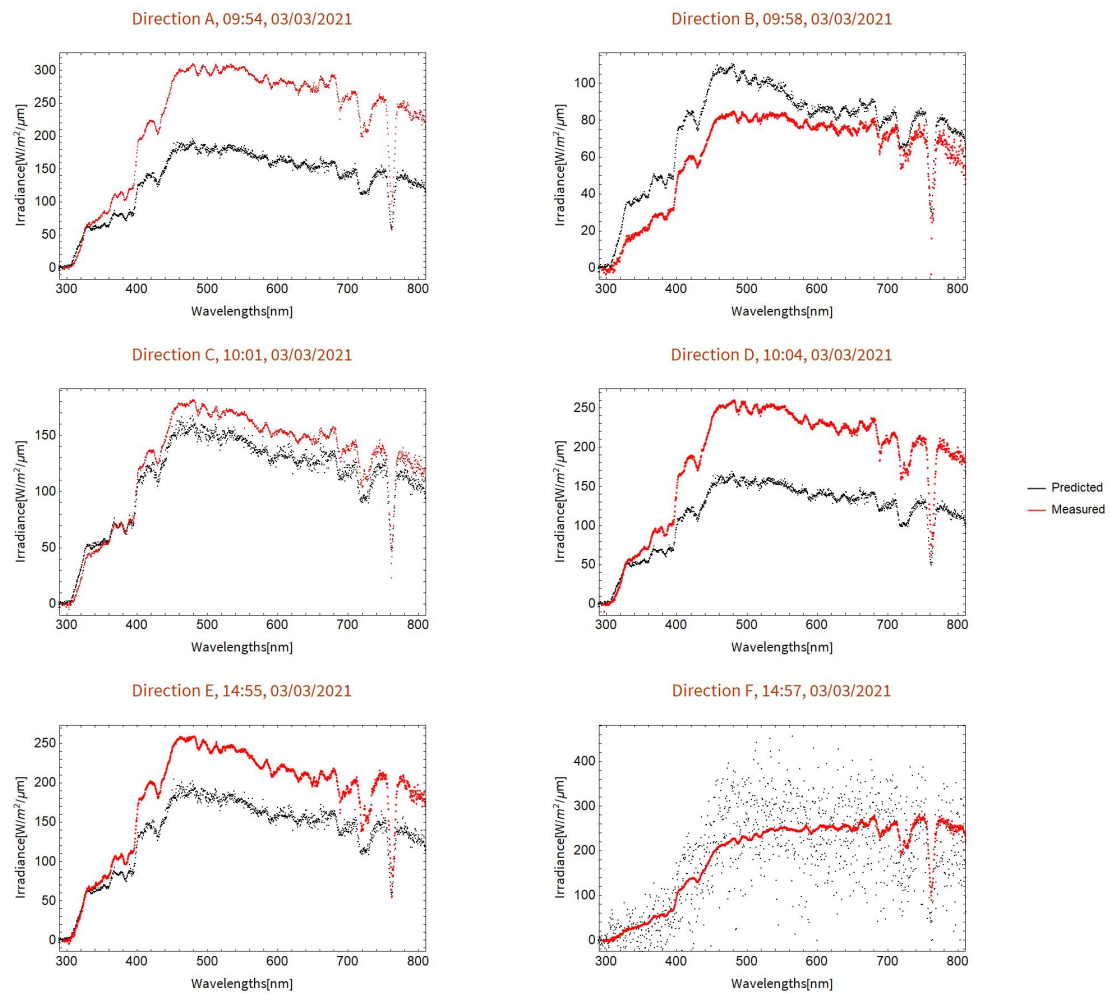


Figure 29: A set of Spectral data comparison in directions A to F on 2021/03/03.



Figure 30: A set of Spectral data comparison in directions G to L on 2021/03/03.

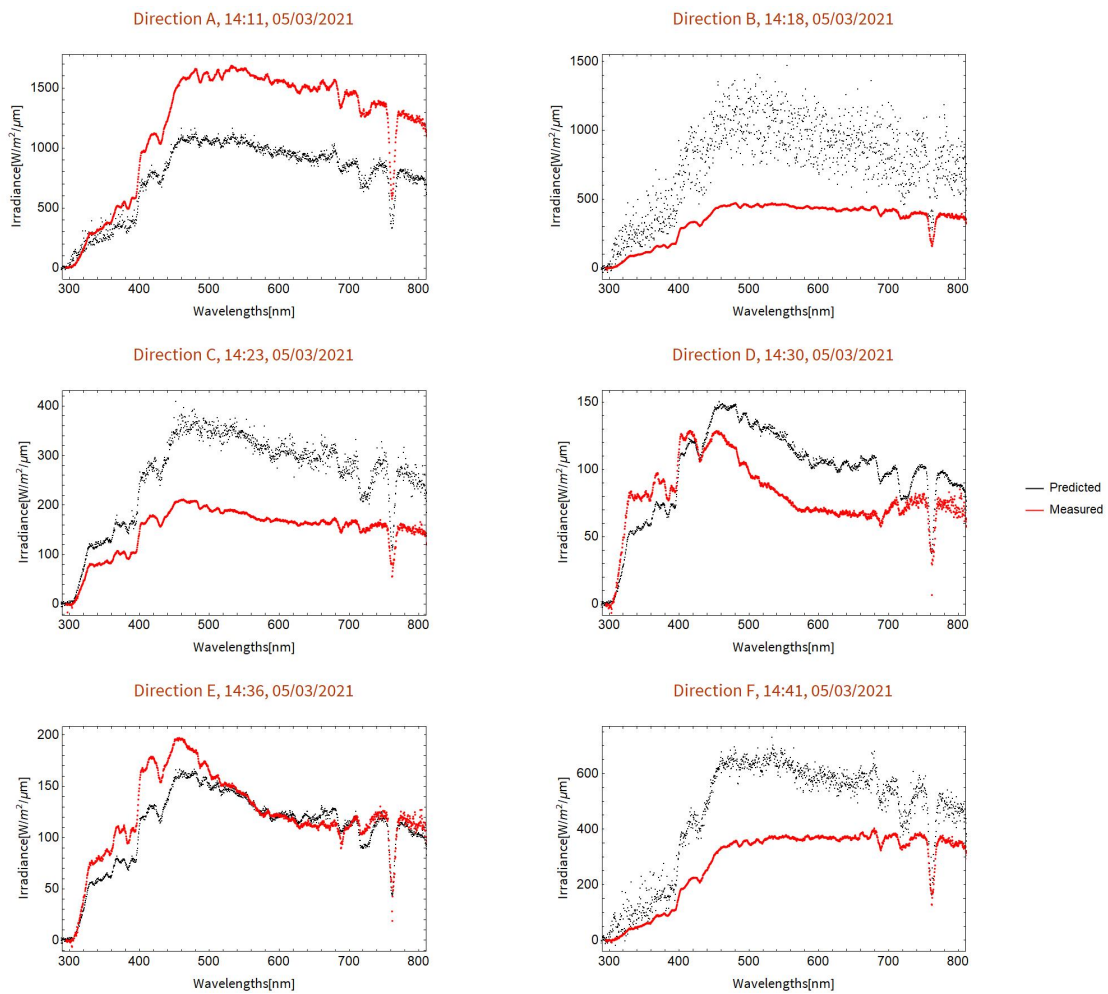


Figure 31: A set of Spectral data comparison in directions A to F on 2021/03/05.

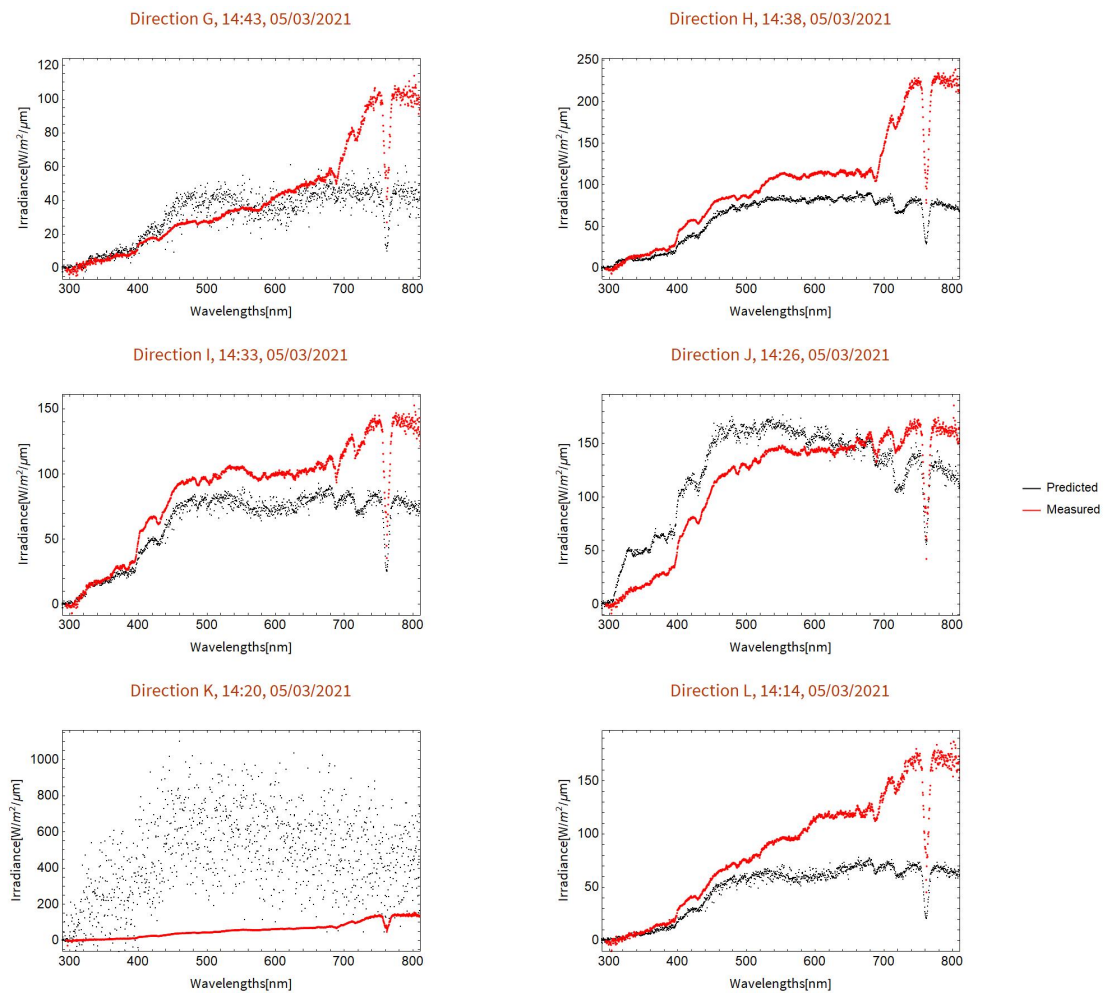


Figure 32: A set of Spectral data comparison in directions G to L on 2021/03/05.

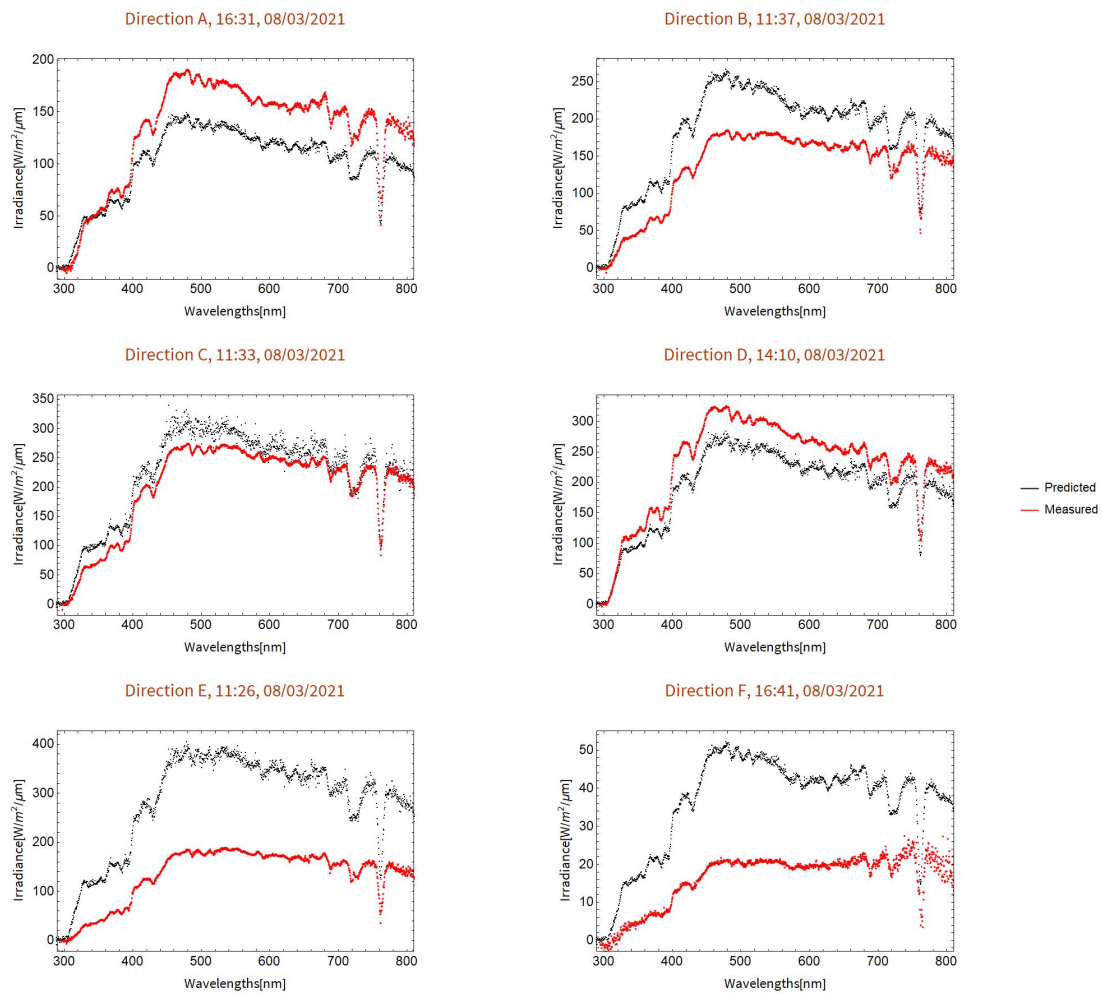


Figure 33: A set of Spectral data comparison in directions A to F on 2021/03/08.

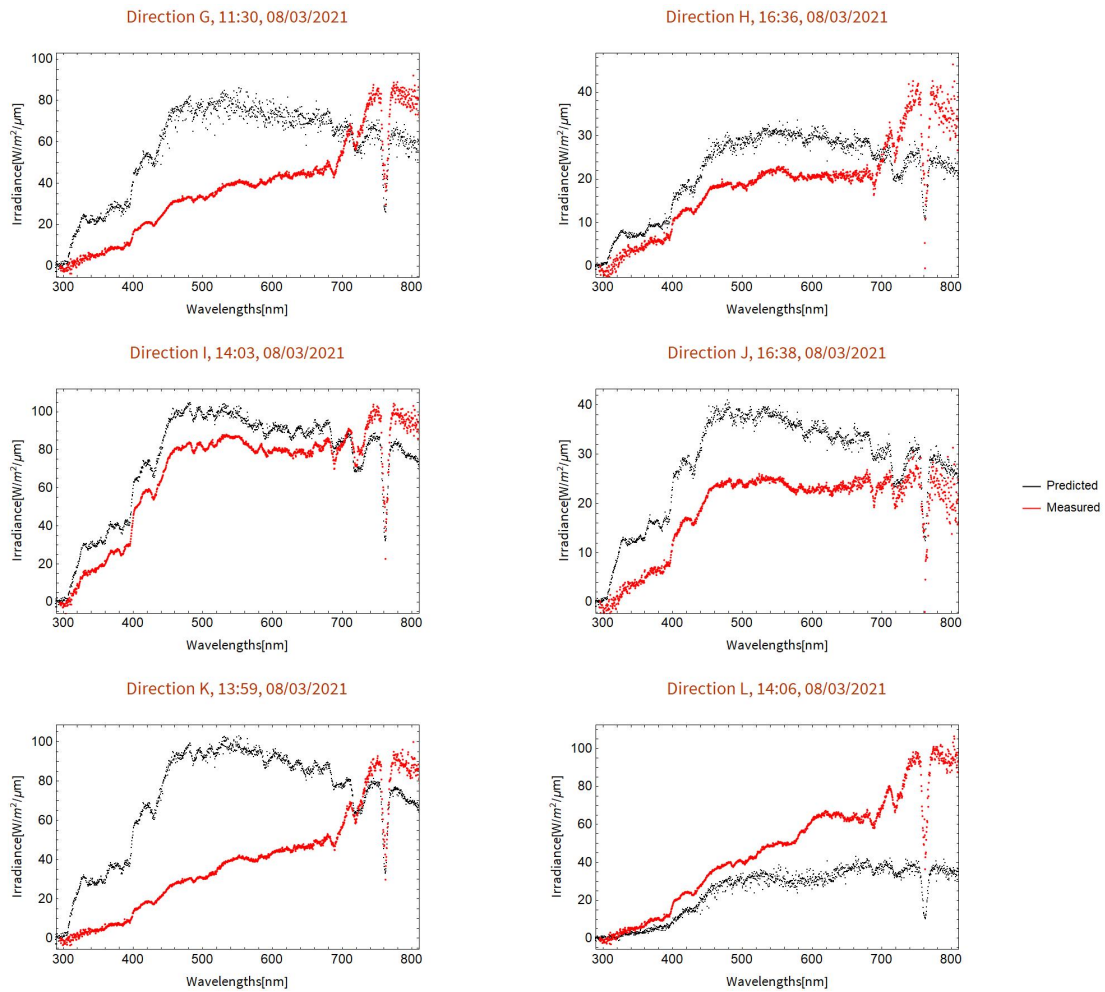


Figure 34: A set of Spectral data comparison in directions G to L on 2021/03/08.

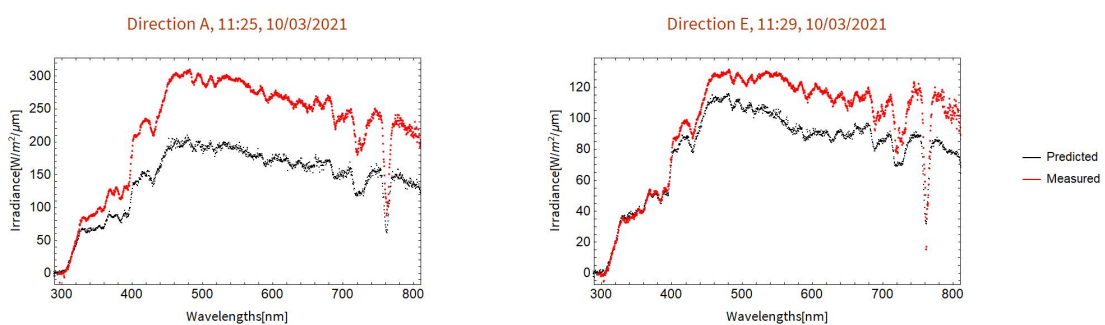


Figure 35: A set of Spectral data comparison in directions A and E on 2021/03/10.

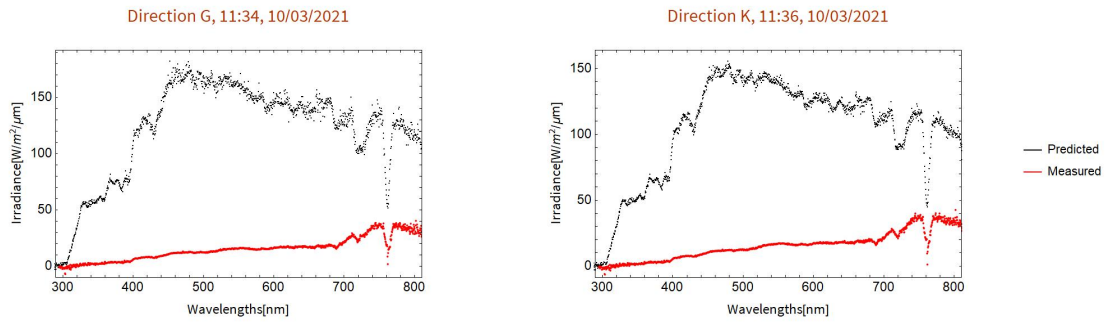


Figure 36: A set of Spectral data comparison in directions G and K on 2021/03/10.

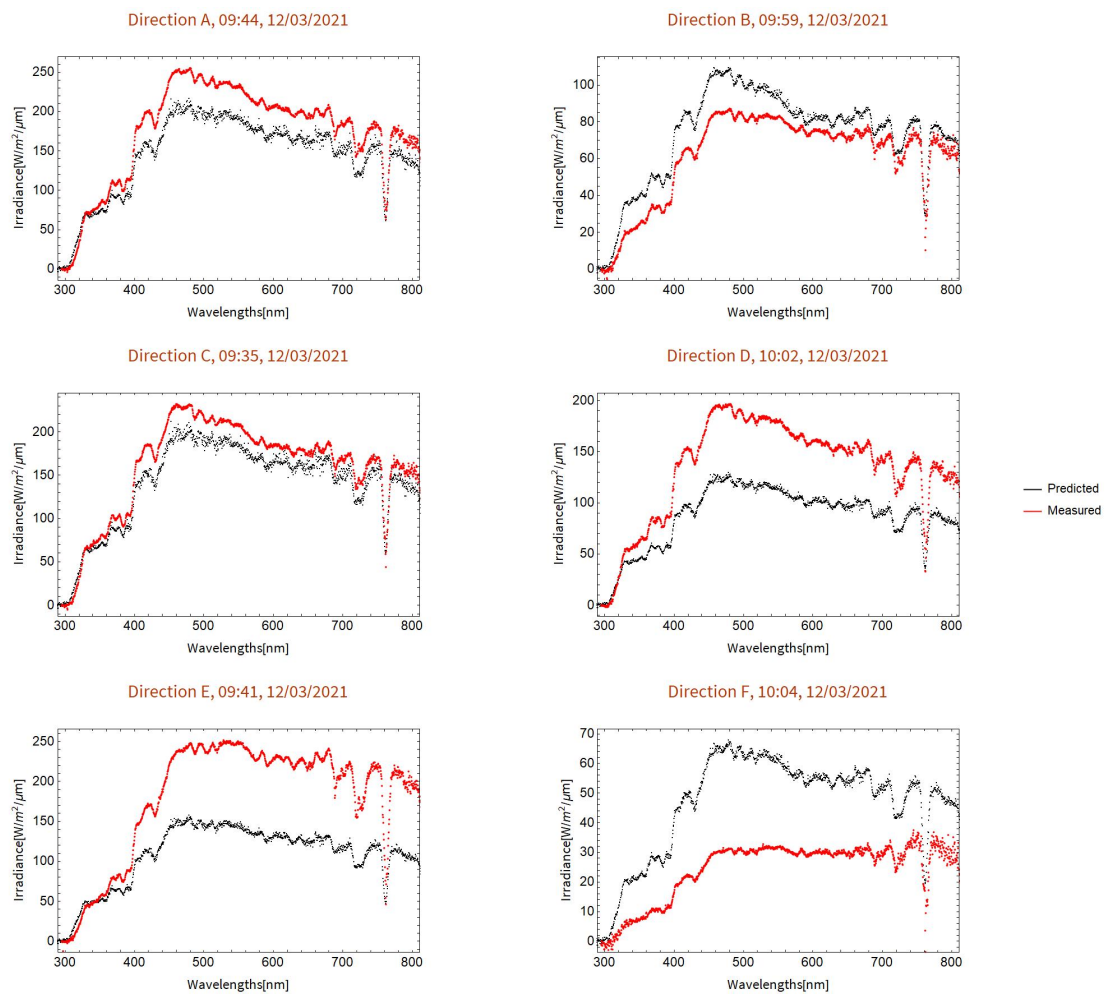


Figure 37: A set of Spectral data comparison in directions A to F on 2021/03/12.

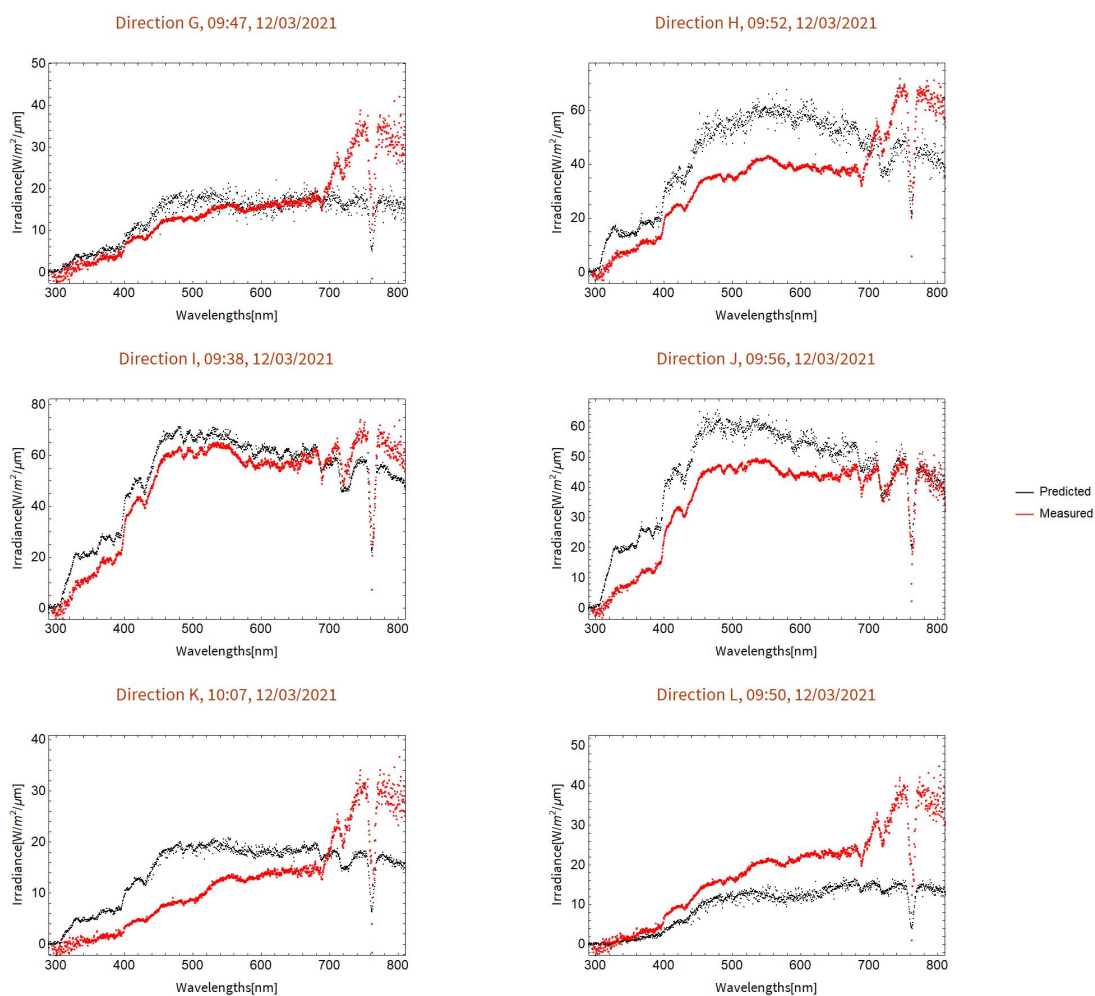


Figure 38: A set of Spectral data comparison in directions G to L on 2021/03/12.

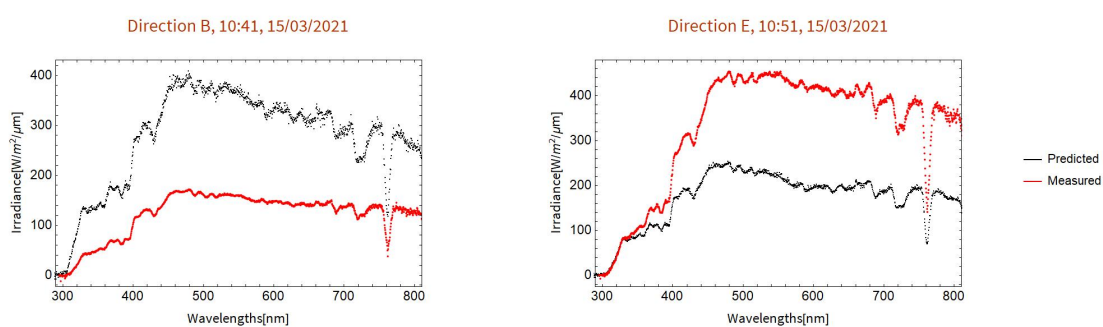


Figure 39: A set of Spectral data comparison in directions B and E on 2021/03/15.

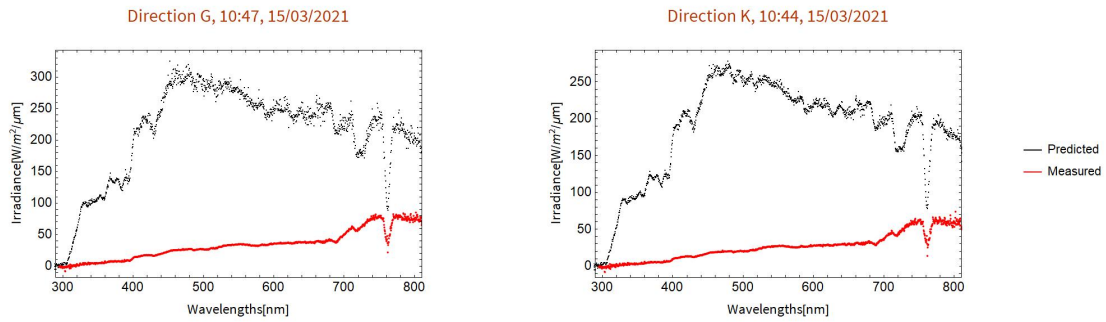


Figure 40: A set of Spectral data comparison in directions G and K on 2021/03/15.

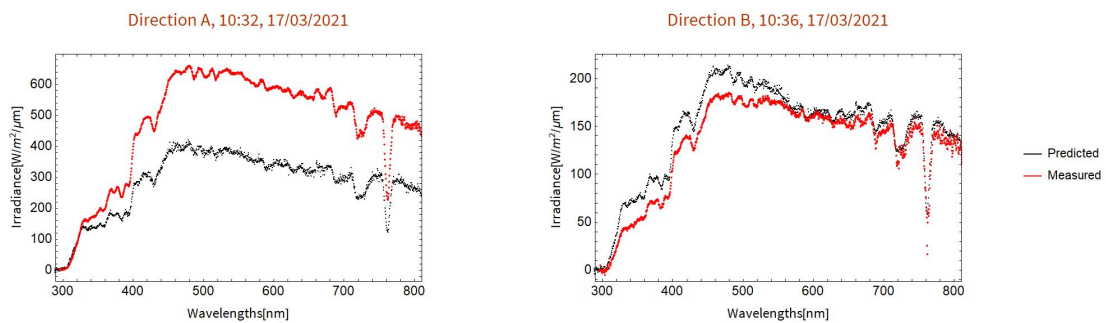


Figure 41: A set of Spectral data comparison in directions A and on 2021/03/17.

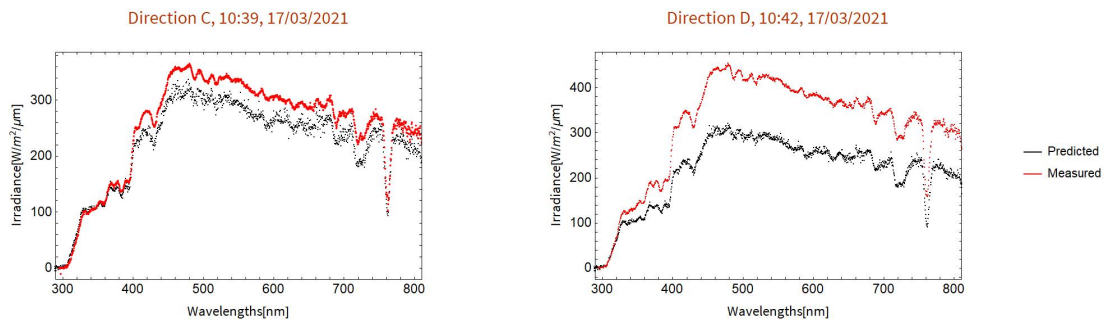


Figure 42: A set of Spectral data comparison in directions C and D on 2021/03/17.

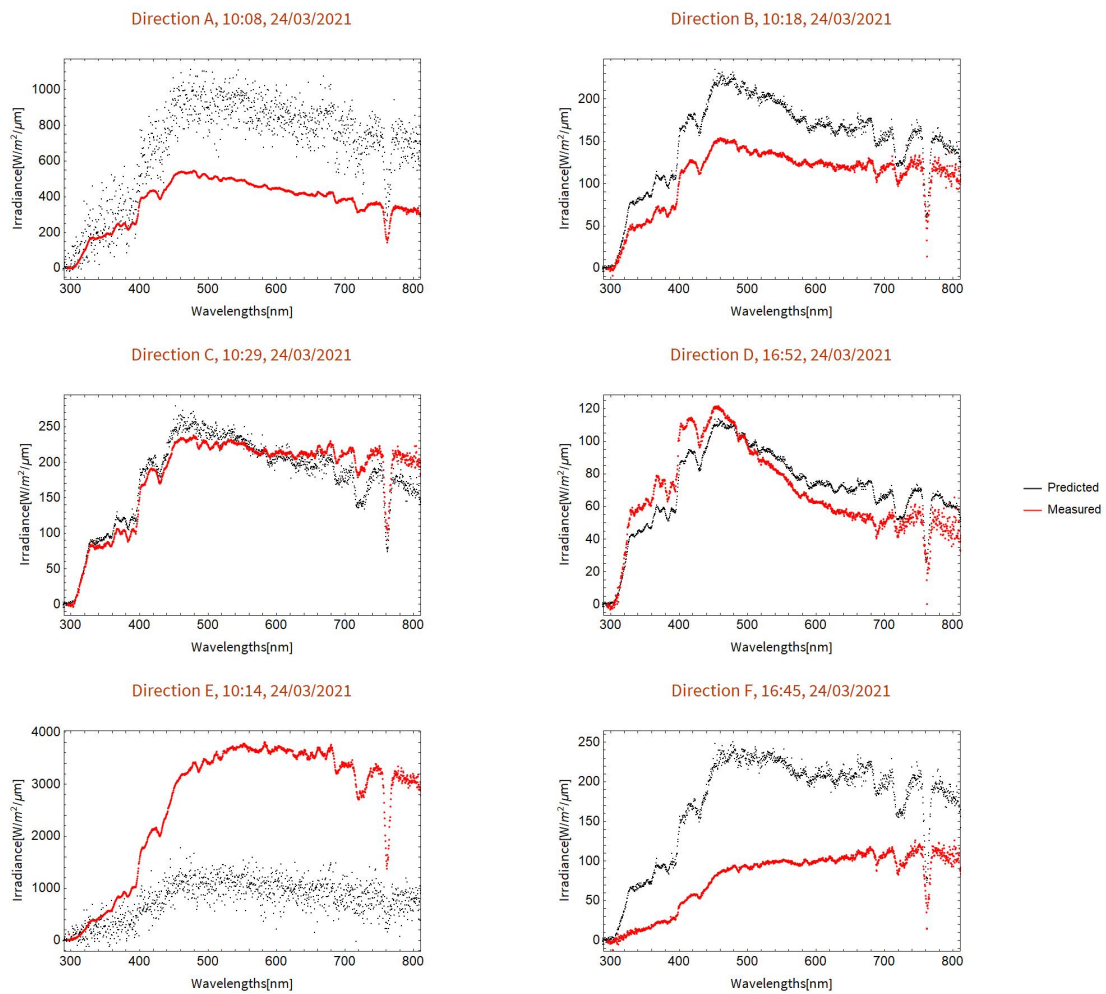


Figure 43: A set of Spectral data comparison in directions A to F on 2021/03/24.

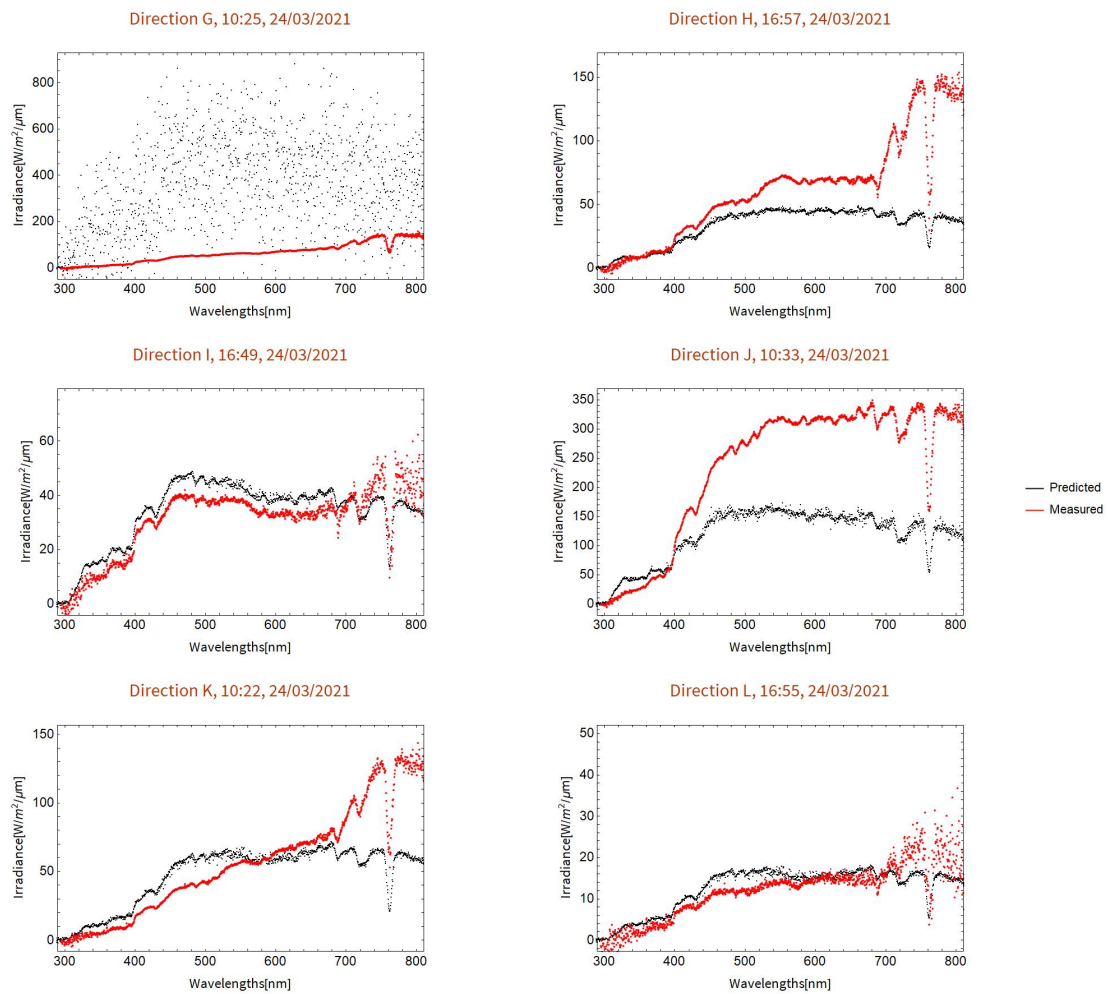


Figure 44: A set of Spectral data comparison in directions G to L on 2021/03/24.

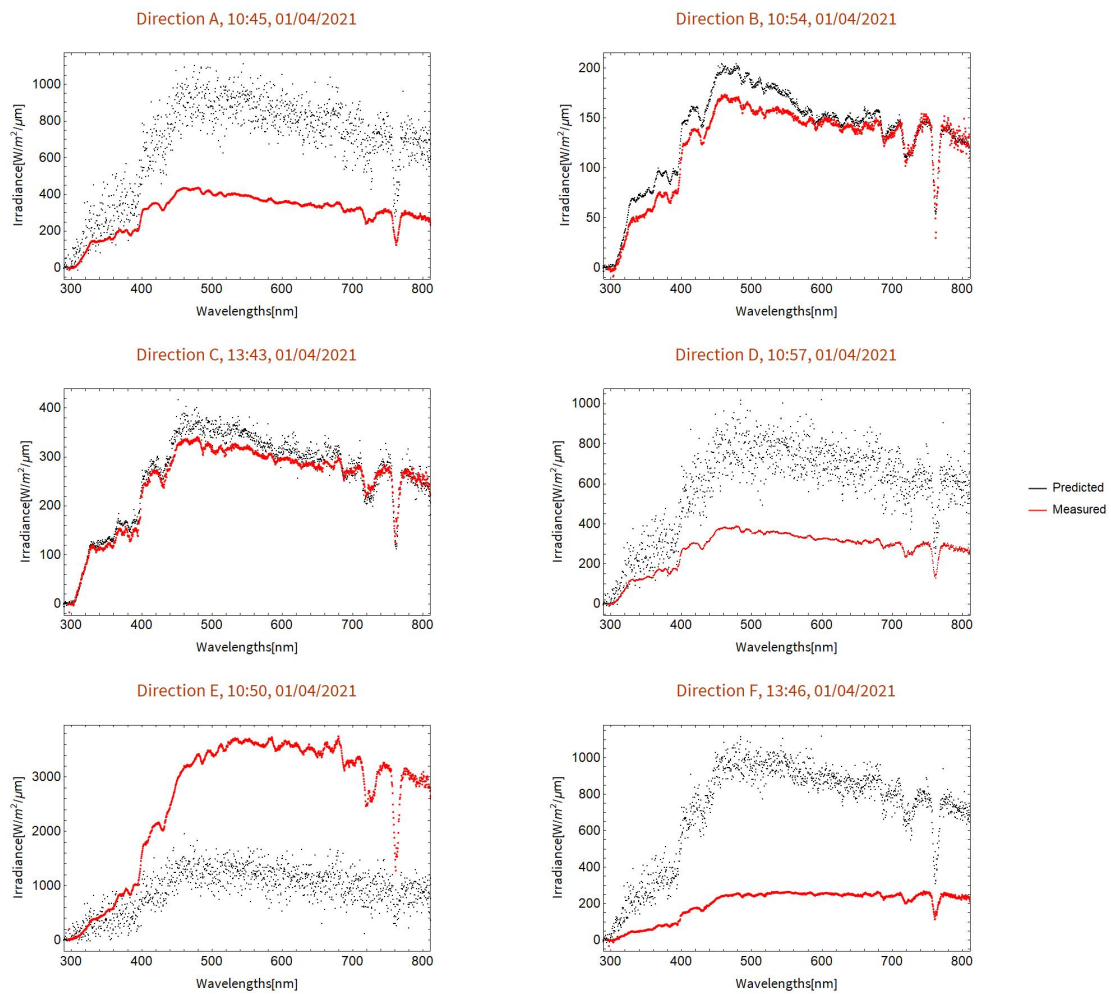


Figure 45: A set of Spectral data comparison in directions A to F on 2021/04/01.

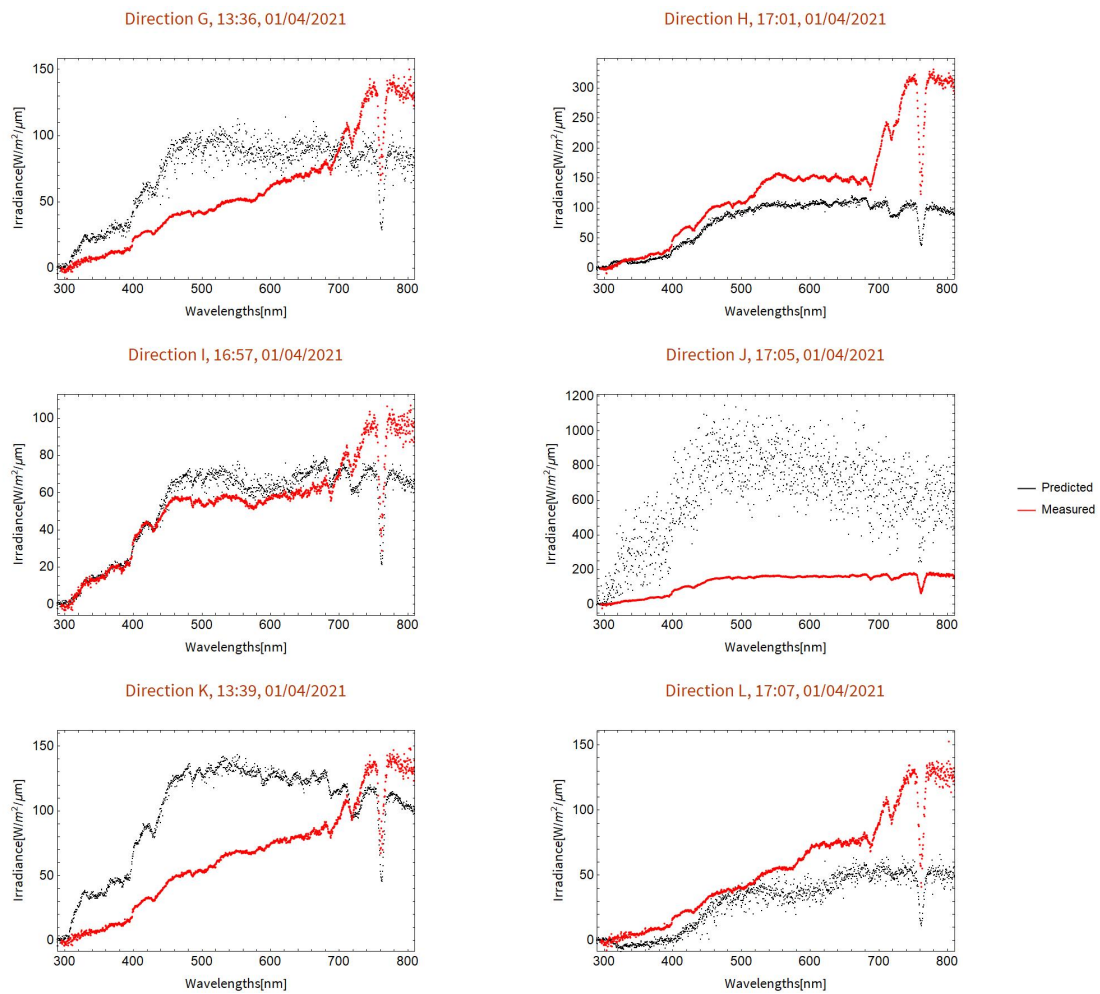


Figure 46: A set of Spectral data comparison in directions G to L on 2021/04/01.

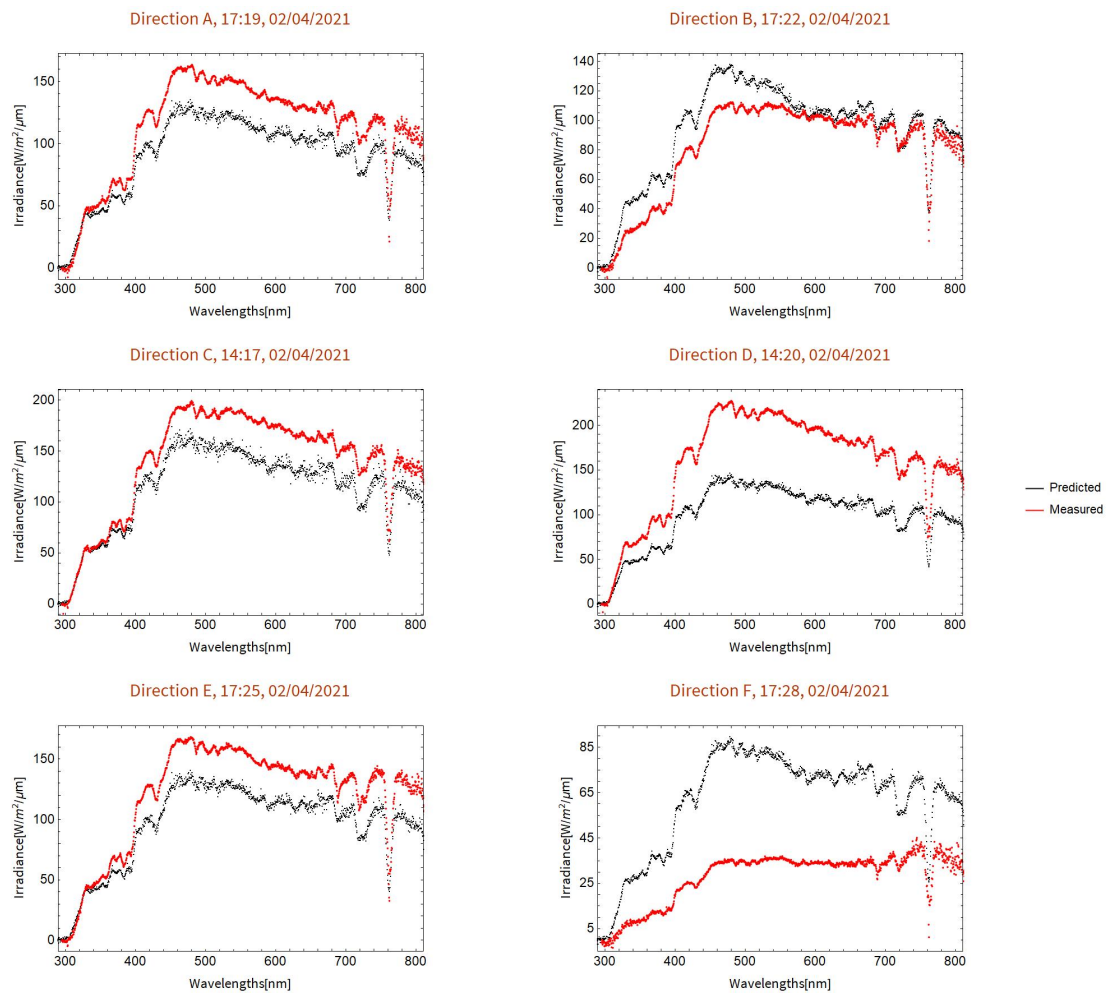


Figure 47: A set of Spectral data comparison in directions A to F on 2021/04/02.



Figure 48: A set of Spectral data comparison in directions G to L on 2021/04/02.

Data Processing Codes

Comments in the code are closed off by the signs (* and *).

Code 1

```

In[*]:= ClearSystemCache[]
ClearAll["Global`*"]

SetDirectory[StringJoin[ParentDirectory[NotebookDirectory[]], "\\Data\\Parameters"]];
constant = Import["wavelength_shift_calibration_constant.mx"];

In[*]:= (*Wavelength Correction applied.*)

In[*]:= PartitionExport[col_, data_, exportdirectory_] := {
  x = data[[col, All]];
  filetype = ToString[First[x[[1]]]];
  filename = StringJoin[filetype, ".mx"];
  SetDirectory[exportdirectory];
  Export[filename, x];
}
AlphabetSelector[j_] := StringJoin[Characters[FileNames["*.mx"]][[j]][[1 ;; 6]]]
Fhandle[x_] := First[{
  directiontypes = DeleteDuplicates[
    Table[AlphabetSelector[1], {1, First[Dimensions[FileNames["*.mx"]]]}]];
  P = First[Dimensions[FileNames[StringJoin[directiontypes[[x]], "*.mx"]]];
  file[x, k_] := FileNames[StringJoin[directiontypes[[x]], "*.mx"]][[k]];
  IntegrationTime = First[Import[file[x, 1]][[3]]];
  dummyfunction[x, m_] :=
    Drop[First[Transpose[N[Import[file[x, m]] / IntegrationTime]], 6];
  A = Total[Table[dummyfunction[x, m], {m, P}] / P
}]
(*This function first finds all the types of data there are
(for this case the different time periods). It then finds how many data
entries there are under the same time and their integration times. It then
averages the data with the number of data entries under the same direction
after they are all divided under their own integration times. The first
6 rows are dropped since they do not have numerical information. At last
all the negative and null data are replaced with 1/Integration Time.*)

```

Figure 49: Part I of the first Code as described in Methodology.

```

In[ ]:= Giant[number_, input_, output_] := {
    SetDirectory[NotebookDirectory[]];
    CreateDirectory["Temp"];
    SetDirectory[input];
    name = FileNames["*.csv"][[number]];
    data = Import[name, "Data"];
    Wavelengths = Drop[data[[All, 1 ;; 1]], 6] - constant;
    Num = Dimensions[data][[2]];
    Parts = First[Partition[data, {1640, 1}]];
    ED = StringJoin[NotebookDirectory[], "Temp"];
    Do[PartitionExport[i, Parts, ED], {i, 2, Num}];
    SetDirectory[StringJoin[NotebookDirectory[], "Temp"]];
    maxiterations = Length[DeleteDuplicates[
        Table[AlphabetSelector[j], {j, First[Dimensions[FileNames["*.mx"]]}]]];
    finaldata = Join[Wavelengths, Table[Fhandle[k], {k, maxiterations}]]', 2];
    header = Join[{Wavelength[nm]}, DeleteDuplicates[
        Table[AlphabetSelector[i], {i, First[Dimensions[FileNames["*.mx"]]}]]];
    finalfinaldata = Join[ArrayReshape[header, {Length[header], 1}], finaldata', 2];
    SetDirectory[output];
    Export[name, finalfinaldata];
    SetDirectory[StringJoin[NotebookDirectory[], "Temp"]];
    DeleteFile[FileNames["*.mx"]];
    SetDirectory[NotebookDirectory[]];
    DeleteDirectory["Temp"];
}

(*This function compiles everything. It first partitions the given
data into a number of columns equal to its column number and exports
them into a temporary directory. It then reads the column titles,
divide data with their respective integration times and combines data
with the same time before it averages them. After that it Joins
all the data with the given wavelength with the given wavelengths
column(Every data is measured with the same spectrometer so they all
share the exact same first column with the same wavelengths). The
data is then exported, with the temporary files deleted.*)

In[ ]:=  $\alpha$  = StringJoin[ParentDirectory[NotebookDirectory[]], "\\Data\\Spectrometer\\Measured"];
 $\beta$  = StringJoin[ParentDirectory[NotebookDirectory[]],
    "\\Data\\Spectrometer\\Wavelength_Calibrated"];
SetDirectory[ $\alpha$ ];
Do[Giant[i,  $\alpha$ ,  $\beta$ ], {i, Length[FileNames["*.csv"]]}]

In[ ]:= ClearSystemCache[]
ClearAll["Global`*"]

```

Figure 50: Part II of the first Code as described in Methodology.

Code 2

```

In[*]:= ClearSystemCache[]
ClearAll["Global`*"]

In[*]:= SetDirectory[StringJoin[ParentDirectory[NotebookDirectory[]], "\\Data\\Parameters"]];
ValidWavelength = Select[Import["Wavelengths.mx"], # < 1091 &];
(*A column of wavelengths of size 2048 that we will use for later interpolation.*)

In[*]:= SetDirectory[
  "C:\\Users\\mychi\\Documents\\AUC\\Capstone\\AMOLF\\Data\\Cleaned_data\\Spectrometer\\
  LAD_New"];
Handle[i_, data_] := First[{
  Firstx = data[[All, 1 ;; 1]];
  y = data[[All, i ;; i]];
  Join[Firstx, y, 2]
}]
(*Takes the first column and some other column of a matrix and joins them*)
name[j_] := FileNames["*.csv"][[j]]
Func[k_] := First[{
  data = Import[name[k], "Data"];
  length = Dimensions[data][[2]];
  Table[Handle[i, data], {i, 2, length}]
}]
EquiInterpolate[data_] := First[{
  f = Interpolation[data];
  (*Create interpolation function from data*)
  y = ArrayReshape[f[ValidWavelength], {Length[ValidWavelength], 1}];
  x = ArrayReshape[ValidWavelength, {Length[ValidWavelength], 1}];
  (*Use wavelength values from LAD
  predictor to get desired counts for later comparison with LAD*)
  Idata = Join[x, y, 2]
}]

```

Figure 51: Part I of the second Code as described in Methodology.

```

Giant[number_, inputdirectory_, outputdirectory_] := {
  SetDirectory[inputdirectory];
  filenumber = number;
  dataname = name[filenumber];
  data = Import[dataname, "Data"];
  test = Table[
    EquiInterpolate[Delete[Func[filenumber][[i]], 1], {i, Dimensions[data][[2]] - 1}];
  Temp = ConstantArray[1, {Length[ValidWavelength] + 1, Dimensions[data][[2]]}];
  Temp[[1, All]] = data[[1, All]];
  Temp[[2 ;; Length[ValidWavelength] + 1, 1]] = ValidWavelength;
  Do[Temp[[2 ;; Length[ValidWavelength] + 1, i + 1]] = test[[i]][[All, 2]],
    {i, Dimensions[data][[2]] - 1}];
  interpolateddata = Dataset[Temp];
  SetDirectory[outputdirectory];
  Export[dataname, interpolateddata]
}
(*Uses all the aforementioned functions for a a construction of a for-
loop to interpolate all the functions.*)

input = StringJoin[ParentDirectory[NotebookDirectory[]],
  "\\Data\\Spectrometer\\Wavelength_Calibrated"];
output = StringJoin[ParentDirectory[NotebookDirectory[]],
  "\\Data\\Spectrometer\\Interpolated"];
SetDirectory[input];
Do[Giant[q, input, output], {q, Length[FileNames["*.csv"]]}]
(*For-loop that detects the number of files and process all of them.*)

In[*]:= ClearSystemCache[]
ClearAll["Global`*"]

```

Figure 52: Part II of the second Code as described in Methodology.

Code 3

```

In[ ]:= ClearSystemCache[]
ClearAll["Global`*"]

In[ ]:= SetDirectory[StringJoin[ParentDirectory[NotebookDirectory[]], "\\Data\\Parameters"]];
CalibrationCurve = Import["Intensity_Calibration_Lines.mx"];

TestFunction[data_, i_, Curve_] := {
  CalibratedData = Drop[data[[All, i]], 1] / Curve
}[[1]]
(*The spectra is divided by the Calibration curve
since the regression lines' function is ADCounts=a*Irradiance.*)
Giant[data_, Curve_] := {
  total = Table[TestFunction[data, i, Curve], {i, 2, Length[data[[1, All]]}]]
}[[1]]

Converter[i_, input_, output_] := {
  SetDirectory[input];
  filename = i;
  Testdata = Import[FileNames["*.csv"]][[filename]];
  name = FileNames["*.csv"][[filename]];
  b = Giant[Testdata, CalibrationCurve];
  wavelengths = Drop[Testdata[[All, 1 ;; 1]], 1];
  fulldata = Drop[Join[wavelengths, b], 2], 16];
  header = ArrayReshape[Testdata[[1, All]], {Length[Testdata[[1, All]]], 1}];
  conclusion = Join[header, fulldata, 2]^T;
  SetDirectory[output];
  Export[name, conclusion]
}
(*Preparation for a for-loop for all the relevant files.*)

input = StringJoin[ParentDirectory[NotebookDirectory[]],
  "\\Data\\Spectrometer\\Interpolated"];
output = StringJoin[ParentDirectory[NotebookDirectory[]],
  "\\Data\\Measured_Spectra_Calibrated"];
SetDirectory[input];
Do[Converter[i, input, output], {i, Length[FileNames["*.csv"]]}]
(*For-Loop that applies the calibration curve to all the relevant spectra.*)

In[ ]:= ClearSystemCache[]
ClearAll["Global`*"]

```

Figure 53: The third Code as described in Methodology.

Code 4

```

In[ ]:= ClearSystemCache[]
ClearAll["Global`*"]

takeline[line_] := {
  tryoutfunction = StringTake[line, 2];
  If[tryoutfunction == "SA" || tryoutfunction == "SB" || tryoutfunction == "SC" ||
    tryoutfunction == "SD" || tryoutfunction == "SE" || tryoutfunction == "SF" ||
    tryoutfunction == "SG" || tryoutfunction == "SH" || tryoutfunction == "SI" ||
    tryoutfunction == "SJ" || tryoutfunction == "SK" || tryoutfunction == "SL",
    f = StringSplit[line, ","];
    For[i = 2, i < 7, i++,
      If[f[[i]] == "00000", f[[i]] = 0];
      outputvalue = {daytime, f[[1]], ToExpression[f[[2]]], ToExpression[f[[3]]],
        ToExpression[f[[4]]], ToExpression[f[[5]]], ToExpression[f[[7]]]};
      date = StringDrop[line, 2];
      If[tryoutfunction == "H", day = StringDrop[date, -9]; t = StringDrop[date, 11];
      daytime = StringJoin[day, ",", t];
      outputvalue = {0, 0, 0, 0, 0, 0, 0};
    ];
  outputvalue[[1]], outputvalue[[2]], outputvalue[[3]],
  outputvalue[[4]], outputvalue[[5]], outputvalue[[6]], outputvalue[[7]]
}

convertImport[text_] := {
  Table[
    takeline[text[[n]]],
    {n, Range[Length[text]]}
  ]
}

(*Jouke's codes on processing the Imported LAD File into something readable.*)
LADImport[filename_, num_, num1_] := {
  file = Import[filename, "Data"];
  cIm = convertImport[file];
  csv = Select[cIm[[1]], # != {0, 0, 0, 0, 0, 0, 0} &];
  DateObject[csv[[12 * num + 1]][[1]], csv[[12 * num + num1]][[3]; 7]]
}

(*Extracts the measured quantities of a particular direction of a particular date
when given the specifications. num_ is determined by the number of minutes
multiplied by 10 (ex: 10:30→num_=30*6=180). num1_ is the direction input;
alphabets A to L is paired with 1 to 12 by ascending order (ex: A→1, B→2,...,L→12).*)
SetDirectory[StringJoin[ParentDirectory[NotebookDirectory[]], "\\Data\\Parameters"]);
lownet = Import["lownet.mx"];
highnet = Import["topnet.mx"];
Wavelength = Import["Wavelengths.mx"];
(*lownet and highnet are the Neural Networks;
the former for intensity values >6000 and the latter the opposite. Wavelength

```

Figure 54: Part I of the fourth Code as described in Methodology.

```

    is the wavelength values paired with the predicted Irradiance values.*)
predict[lad_] :=
If[lad[[1]] ≥ 6000,
  out = highnet["TrainedNet"][lad],
  out = lownet["TrainedNet"][lad]
]
(*Determine when to use which Neural Network.*/)
Predicteddata[xdata_, ydata_] := First[{
  x1 = ArrayReshape[xdata, {2048, 1}];
  y1 = ArrayReshape[ydata, {2048, 1}];
  Data1 = Join[x1, y1, 2]
}]
Data[data_, j_] := {
  directionnumber = j;
  hour = StringJoin[Characters[data[[1, directionnumber]]][[3 ;; 4]]];
  minute = StringJoin[Characters[data[[1, directionnumber]]][[5 ;; 6]]];
  LADsource = StringJoin[ParentDirectory[NotebookDirectory[]], "\\Data\\LAD"];
  SetDirectory[LADsource];
  FileNames[StringJoin[Dates, "_", hour, "*.txt"]];
  SetDirectory[LADsource];
  RawdataI = LADImport[FileNames[StringJoin[Dates, "_", hour, "*.txt"]][[1]],
    ToExpression[minute] + 6, directionnumber - 1];
  RawdataII = LADImport[FileNames[StringJoin[Dates, "_", hour, "*.txt"]][[1]],
    ToExpression[minute] + 6 + 1, directionnumber - 1];
  RawdataIII = LADImport[FileNames[StringJoin[Dates, "_", hour, "*.txt"]][[1]],
    ToExpression[minute] + 6 + 2, directionnumber - 1];
  RawdataIV = LADImport[FileNames[StringJoin[Dates, "_", hour, "*.txt"]][[1]],
    ToExpression[minute] + 6 + 3, directionnumber - 1];
  RawdataV = LADImport[FileNames[StringJoin[Dates, "_", hour, "*.txt"]][[1]],
    ToExpression[minute] + 6 + 4, directionnumber - 1];
  RawdataVI = LADImport[FileNames[StringJoin[Dates, "_", hour, "*.txt"]][[1]],
    ToExpression[minute] + 6 + 5, directionnumber - 1];
  Averageddata = (predict[RawdataI[[2]]] + predict[RawdataII[[2]]] +
    predict[RawdataIII[[2]]] + predict[RawdataIV[[2]]] +
    predict[RawdataV[[2]]] + predict[RawdataVI[[2]]]) / 6
}[[1]]
(*Takes a reference spectrum file,
extracts the relevant timestamps and uses them to find the
right LAD file and data entries of the right minute. The LAD data are
then inputted into the NN. The obtained spectra are then averaged.*/)

```

Figure 55: Part II of the fourth Code as described in Methodology.

Code 5

```

In[24]~ ClearSystemCache[]
ClearAll["Global`*"]

In[25]~ (*This code's purpose is to plot Predicted and Reference spectra
of the same time. All sets from the same date is put into 1 grid.*)

In[622]~ Comparer[i_, j_] := {
  SetDirectory[
    StringJoin[ParentDirectory[NotebookDirectory[]], "\\Data\\Predicted_Spectra"];
    name = FileNames["*.csv"][[i]];
    LADI = Import[name];

    SetDirectory[StringJoin[
      ParentDirectory[NotebookDirectory[]], "\\Data\\Measured_Spectra_Calibrated"];
    SpecI = Import[name];
    (*Takes Predicted (LADI) and Reference (SpecI) spectra measured in the same time.*)
    Dates = Drop[Characters[name], -4];
    time = Drop[Characters[ToString[SpecI[[1, j]]], 2];
    direction = StringJoin[Characters[ToString[SpecI[[1, j]]]][[1]]];

    ymax = If[Max[Drop[LADI[[All, j]], 1]] > Max[Drop[Drop[SpecI[[All, j]], 1], -465]],
      Max[Drop[LADI[[All, j]], 1]], Max[Drop[Drop[SpecI[[All, j]], 1], -465]]];
    plotA = ListPlot[Drop[LADI[[All, {1, j}]], 1], PlotStyle -> Black];
    plotB = ListPlot[Drop[SpecI[[All, {1, j}]], 1], PlotStyle -> Red];
    combiplotA =
      Show[plotA, plotB, PlotRange -> {{300, 800}, {0, ymax}}, Frame -> True, FrameLabel ->
        {Style["Wavelengths[nm]", "Text"], Style["Irradiance[W/m2/μm]", "Text"]},
        FrameStyle -> Directive[Black, 14], PlotLabel -> Style[StringJoin["Direction ",
          direction, ", ", time[[1 ;; 2]], ":", time[[3 ;; 4]], ", ", Dates[[7 ;; 8]], "/",
          Dates[[5 ;; 6]], "/", Dates[[1 ;; 4]], "Subsubsection"], ImageSize -> Medium];

    combiplotA
  }[[1]]
  (*Takes 1 Predicted and Measured spectra and plot them together.*)

```

Figure 56: Part I of the fifth Code as described in Methodology.

```

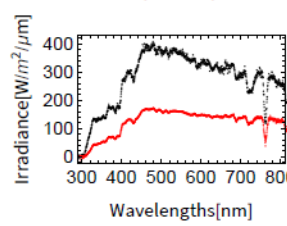
In[624]:= ForLoop[k_] := {
  SetDirectory[StringJoin[
    ParentDirectory[NotebookDirectory[]], "\\Data\\Measured_Spectra_Calibrated"];
  temp = Import[FileNames["*.csv"]][[k]];
  graphs = Table[Comparer[k, 1], {1, 2, Length[temp[[1, All]]}];
  output1 = Legended[GraphicsGrid[
    ArrayReshape[graphs[[1 ;; (Length[graphs] / 2)]]], {(Length[graphs] / 4), 2}],
    ImageSize -> Full], LineLegend[{Black, Red}, {"Predicted", "Measured"}]];
  output2 = Legended[GraphicsGrid[ArrayReshape[
    graphs[(Length[graphs] / 2 + 1) ;; Length[graphs]]], {(Length[graphs] / 4), 2}],
    ImageSize -> Full], LineLegend[{Black, Red}, {"Predicted", "Measured"}]];
  SetDirectory[StringJoin[ParentDirectory[NotebookDirectory[]],
    "\\Pictures\\Data Processing"];
  Export[StringJoin["Predicted_VS_Measured_", Dates, "_I", ".jpg"], output1];
  Export[StringJoin["Predicted_VS_Measured_", Dates, "_II", ".jpg"], output2];
  output1, output2
}
(*For-Loop that plots all the sets of Predicted and Measured Spectra. Put
all sets that are from the same day into one grid, and exports them.*)

In[642]:= test = ForLoop[10];

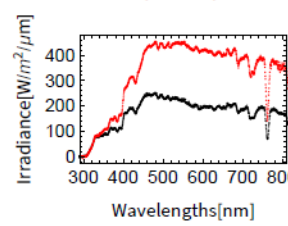
In[641]:= test[[1]]

```

Direction B, 10:41, 15/03/2



Direction E, 10:51, 15/03/2



```

SetDirectory[
  StringJoin[ParentDirectory[NotebookDirectory[]], "\\Pictures\\Data Processing"];
Export[StringJoin["Predicted_VS_Measured_", Dates, "_I", ".jpg"], output1]

In[39]:= ClearSystemCache[]
ClearAll["Global`*"]

```

Figure 57: Part II of the fifth Code as described in Methodology. An example of exported images is included.

References

¹IEA, *Renewables 2020*, (2020) <https://www.iea.org/reports/renewables-2020> (visited on 02/24/2021).

²F. Ise, “Photovoltaics report”, Freiburg, Sept. 20, 2020.

³TÜV Rheinland Energy, *Graph_performance_characteristics_of_bifacial_pv_modules--tuv_rheinland-energy_web-2.jpg (JPEG image, 1183 × 656 pixels)*, (July 12, 2018) https://www.kippzonen.com/data/images/Graph_Performance_characteristics_of_bifacial_PV_modules--TUV_Rheinland_Energy--web-2.jpg (visited on 02/25/2021).

⁴C. D. Rodríguez-Gallegos, M. Bieri, O. Gandhi, J. P. Singh, T. Reindl, and S. K. Panda, “Monofacial vs bifacial si-based PV modules: which one is more cost-effective?”, *Solar Energy* **176**, 412–438 (2018).

⁵J. Conijn and A. van Hemert, “Scientifically measuring light diffuseness in a cost-effective way”, 19.

⁶T. Maassen, W. Sinke, and B. Ehrler, “The effects of light diffuseness on the outdoor performance of thin film solar cells”, PhD thesis (University of Amsterdam, Amsterdam, July 2017), 50 pp.

⁷M. Kersten, F. Bradbury, and E. Llado, “Outdoor solar cell performance: improving a diffuseness measurement tool”, PhD thesis (Amsterdam University College, Amsterdam, May 30, 2018), 43 pp.

⁸A. Pollastri, “A novel multi-directional light detector for modelling the cost-efficiency benefits of bi-facial solar panels”, PhD thesis (Amsterdam University College, Amsterdam, June 2, 2019), 57 pp.

⁹M. Blum, “Utilizing machine learning to predict the solar spectrum from all directions”, PhD thesis (University of Groningen, Groningen, Oct. 27, 2020), 29 pp.

¹⁰*Unmounted n-BK7 ground glass diffusers*, https://www.thorlabs.com/newgrouppage9.cfm?objectgroup_id=1132 (visited on 05/29/2021).

¹¹*MS-711 spectroradiometer — EKO instruments*, <https://eko-eu.com/products/solar-energy/spectroradiometers/ms-711-spectroradiometer> (visited on 05/29/2021).