Multispectral Deep Learning Material Classification for Thermal Imaging

By

Noah James Holliger

Thesis

Submitted to the Faculty of the

Graduate School of Vanderbilt University

in partial fulfillment of the requirements

for the degree of

MASTER OF SCIENCE

in

Interdisciplinary Materials Science

December 17, 2022

Nashville, Tennessee

Approved:

D. Greg Walker, Ph.D.

Soheil Kolouri, Ph.D.

Joshua David Caldwell, Ph.D.

ACKNOWLEDGEMENTS

First, I would like to express gratitude to my research advisor, Prof. Greg Walker, for his guidance, support, and facilitation in my research endeavors during my time at Vanderbilt University. I deeply appreciated the academic freedom Greg afforded me and his encouragement to pursue research avenues that interest me most. At every pivot of my intellectual interests Greg stood behind my decisions in hopes that I find a field of work which provides me with true happiness. Upon informing him of my desire to exit the Ph.D. program to take up a career outside materials science, Greg had one response: "How can I help?" This is truly the sign of a great mentor. I could not be more appreciative of his understanding and support in the pursuit of my career goals.

I would also like to thank my group mate Bradly Baer for his contributions and insight during our weekly group meetings and beyond. Furthermore, thank you to Sarah Ross, program manager at the Vanderbilt Institute of Nanoscale Science and Engineering (VINSE), for her unrelenting efforts to ensure the needs of all VINSE students are met. Her superb organization had a profoundly positive impact on my time here at Vanderbilt and VINSE as a whole.

Thank you to my longtime friend and colleague Marcel Chlupsa for helping to edit this thesis. I wish him the best of luck in the remainder his Ph.D. studies in the Materials Science and Engineering program at the University of Michigan.

Finally, I'd like to thank my family for their unconditional love and support throughout my graduate school experience. I never would have made it this far in my academic career without their help.

TABLE OF CONTENTS

Section Page

LIST OF TABLES

LIST OF FIGURES

Figure Page

CHAPTER I

INTRODUCTION

Thermal imaging is a vital technology to a diverse range of application spaces such as autonomous vehicles, military targeting and surveillance, firefighting, and physiological evaluation [1-4]. Therefore, bolstering the accuracy (by efficient means) of thermal cameras is crucial to further development of these application spaces. In this chapter, we will discuss the current state of thermal imaging and address how to improve it. First, we will consider the fundamentals of the technology in its current form. Next, we will focus on why imaging various emissivities presents a pressing problem in thermal imaging. Then, we will detail how deep learning offers a solution to this problem. Finally, we will review how multispectral thermal imaging enables deep learning to provide an improved solution.

1.1 Fundamentals of Thermal Imaging

A thermal camera captures radiation from a specific spectral range within the far-infrared (FIR) spectrum and constructs a heat map of the imaged scene corresponding to the intensity of the radiation captured by each pixel in the thermal camera. This section will break down how the spectral range is chosen, the radiation is captured and measured, and the temperature calculation is performed.

1.1.1 Optimal spectral range for thermal imaging

Most thermal cameras utilize a spectral range from 7.5 μm to 14 μm [5]. There are two key reasons for the selection of this spectral range. First, the composition of Earth's atmospheric gases gives rise to several atmospheric transmission windows. These windows offer high transmission of electromagnetic radiation due to the low absorptivity of common atmospheric gases within the spectral range of the windows. Any device which relies on capturing electromagnetic radiation that travels long distances through the atmospheric medium must consider these windows. If such a device were designed to only capture radiation in spectral ranges falling outside any atmospheric windows, it would largely fail to capture radiation from the desired source and instead capture spontaneous radiative emissions from the atmosphere in the case of long-range imaging. Thus, in the design of a thermal camera, it is imperative that the range of radiation capture falls within an atmospheric window. There are several atmospheric windows within the infrared (IR) regime, however largest window stretches from 7.5 μm to 14 μm as seen in Figure 1 [6].

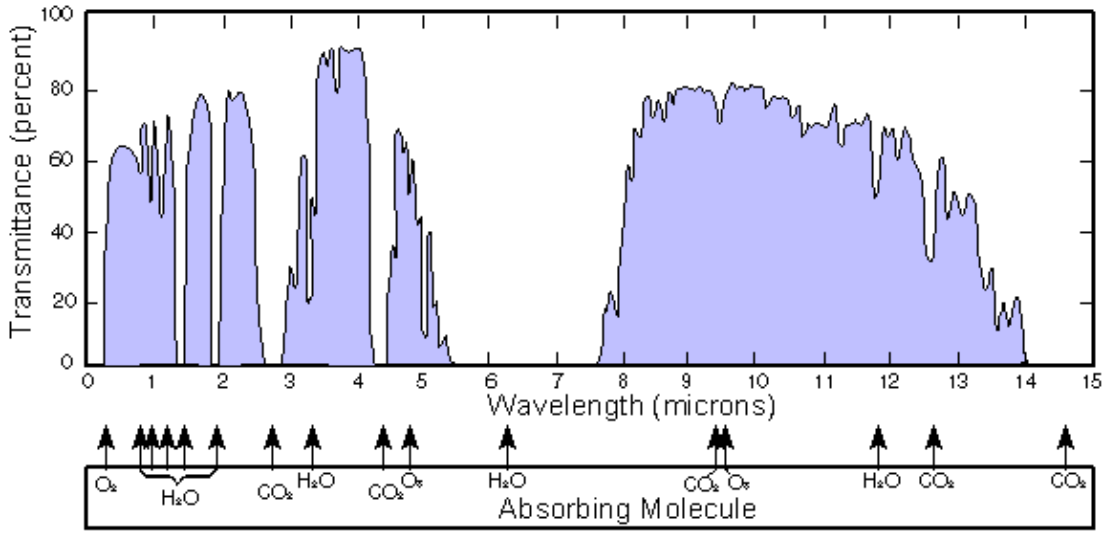

Figure 1: Atmospheric transmittance of electromagnetic radiation in the wavelength range of 0 μm to 15 μm.

The second primary factor in determining the optimal spectral range for a thermal camera is consideration of the spectral and temperature dependent intensity of radiation emitted from the source. At this point, Planck's blackbody radiation distribution, which is seen in Equation 1, must be analyzed [7].

$$E_B(\lambda, T) = \frac{2\pi hc^2}{n^2\lambda^5} \frac{1}{e^{\frac{hc}{n\lambda K_b T}} - 1} \qquad (1)$$

Here c is defined as the speed of light in a vacuum, n is the refractive index ($n \approx 1$ in atmospheric conditions), λ is the wavelength of light emitted, and T is the temperature of the object. Values of h and K_b are given as Planck's and Boltzmann's constants, respectively. Ideally, the spectral range chosen should contain the maximum of this function at the temperatures most likely to be measured. Choosing a spectral range which satisfies this criterion does not require as low of a sensitivity threshold for the measurement device as does a spectral range which produces less emissive power. As a result, this design choice yields greater manufacturing freedom, thus enabling a less expensive production cost and a more affordable product to consumers. Observing Figure 2 shows that at ambient temperatures (300 K), the maximum of Planck's blackbody distribution resides at nearly the exact center of the 7.5 μm to 14 μm atmospheric window [8].

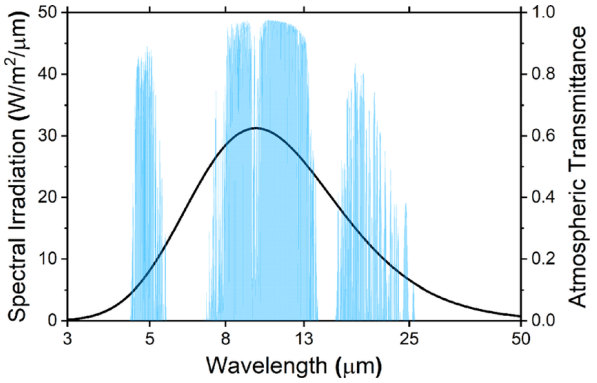

Figure 2: Planck's law at 300 K in the IR region (black) plotted over the FIR atmospheric transmission windows (blue).

It should be noted that as an object's temperature increases, the peak irradiance will shift towards lower wavelengths. The peak irradiance is approximated by Wein's law which is seen in Equation 2 [7].

$$\lambda_{max} = \frac{b}{T} \qquad (2)$$

The equation defines λ_{max} as the wavelength at which a blackbody produces the peak irradiance, b as Wein's displacement constant, and T as temperature. Using Wein's law, the approximate temperature at which a blackbody's peak irradiance wavelength falls below 7.5 μm is 386 K and above. This is not a concern as the total irradiance within the 7.5 μm to 14 μm spectral range will still be greater than at 300 K, thus the sensitivity threshold of the measurement device is still met. However, if temperatures decrease below 207 K the peak irradiance wavelength will increase past 14 μm. Coupled with low total irradiance, utilizing only the 7.5 μm to 14 μm atmospheric window requires a very low sensitivity threshold. For these reasons, low temperature measurements require a more specialized thermal camera. Nevertheless, in most use cases a thermal camera with a 7.5 μm to 14 μm spectral range is optimal.

1.1.2 Capturing and measuring IR radiation for thermal imaging

There are two main types of IR detectors which a thermal camera may employ: cooled quantum detectors and microbolometer detectors. The cooled quantum detector relies on cooling low-bandgap materials (such as InSb) to temperatures low enough that the photoelectric effect can be used to measure IR radiation [5]. This is the type of detector that would be used in the specialized low temperature thermal camera mentioned in the previous section. The vast majority

of thermal cameras operate using a detector called a microbolometer. This subsection will focus on the microbolometer detector.

A microbolometer is comprised of an IR absorbing material, a temperature sensing material, a reflector, two metal contacts, and a readout integrated circuit (ROIC) [9]. The absorbing material is a thin-film metal deposited on top of the sensing material. The sensing material quickly reaches thermal equilibrium with the absorbing material contacting it. This membrane is constructed to have the absorbing material approximately 2.5 μm above the reflecting layer. The spacing forms a λ/4 resonator which maximizes and isolates absorption of electromagnetic radiation in the desired 7.5 μm to 14 μm spectral range. The membrane's sensing material is designed with long thin bridges that connect to the metal contacts suspending the membrane. These bridges provide thermal insulation between the membrane and the ROIC. The metal contacts connect to the ROIC, which enables measuring the resistance of the sensing material. Each microbolometer acts as a single pixel within the thermal camera, thus they are fabricated into an array with dimensions of the desired resolution. SEM images depicting the structure of a microbolometer array can be seen in Figure 3 [10].

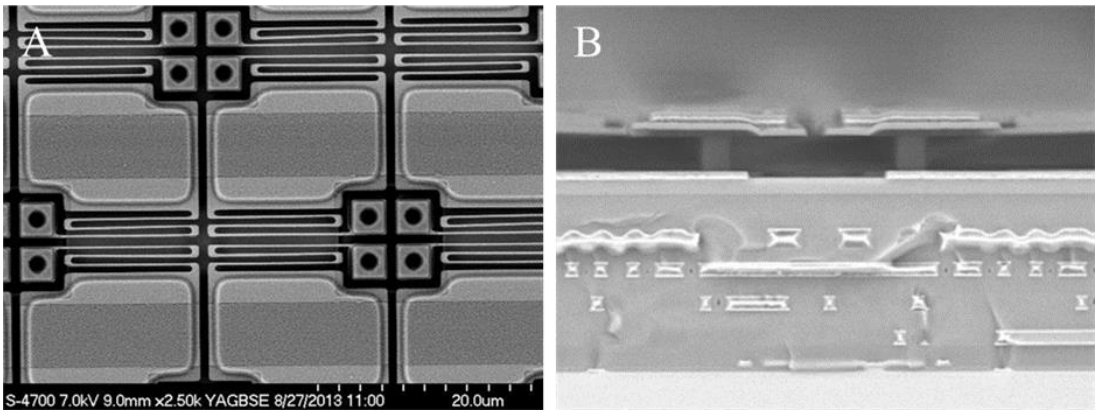

Figure 3: (**A**) Top-view of a microbolometer array and (**B**) side-view of two microbolometers with the ROIC in view.

The thin-film metal in the absorbing layer preferably has a low heat capacity to produce a large temperature response to absorbed radiation. Ideally, it should also possess a high absorptivity in the FIR region. The absorbing layer is commonly made with Ti, NiCr, or TiNb [11]. A sensing material should have a high temperature coefficient of resistance to maximize the sensitivity of the device. It should also have a low thermal conductivity to provide as much thermal insulation between the membrane and the ROIC as possible. Given these desired properties, the sensing material is often constructed using amorphous Si or V_2O_5 [12].

Once the microbolometer array is fabricated, it is housed within a vacuum package that utilizes an IR transmissive cap that allows electromagnetic radiation to pass through to the array. The vacuum packaging prevents heat exchange with the atmosphere. The packaged array is fitted to a camera with an IR transmissive lens (often Ge). The lens focuses incoming radiation onto the microbolometer array. With the camera assembled, each pixel in the array is calibrated to establish the relationship between the measured microbolometer resistance and the incident detectable radiant power flux [13]. The relationship is a curve fitted equation which estimates the total detectable radiant power flux, Q_{tot}, as a function of the measured resistance, R, for some pixel, *i*, as seen in Equation 3.

$$Q_{tot} = f_i(R) \tag{3}$$

1.1.3 Solving the object temperature

There are three sources that contribute towards the total detectable radiation incident with the camera upon taking a measurement. The first source is radiation emitted from the imaged object. The radiative power flux emitted by the object that is detected by the camera is derived using the Stefan-Boltzmann Law as seen in Equation 4 [7].

$$Q_{obj} = f(\lambda_1 T_{obj}, \lambda_2 T_{obj})\varepsilon\tau\sigma T_{obj}^4 \quad (4)$$

In this equation, σ is the Stefan-Boltzmann constant and T_{obj} is the temperature of the object. The emissivity of the object is ε and the transmittance of the atmosphere between the object and the camera is τ, with both evaluated for the spectral range of the camera, λ_1 to λ_2. The function f(λ_1 T_{obj}, λ_2 T_{obj}) is the fraction of the blackbody emissive power emitted in the spectral range from λ_1 to λ_2. This function is evaluated using Equation 5 [7].

$$f(\lambda_1 T, \lambda_2 T) = \frac{\int_{\lambda_1}^{\lambda_2} E_B(\lambda,T)d\lambda}{\int_0^{\infty} E_B(\lambda,T)d\lambda} = \frac{15}{\pi^4}\int_{C_1/\lambda_1 T}^{C_1/\lambda_2 T} \frac{\xi^3 d\xi}{e^{\xi}-1} \quad (5)$$

The constant C_1 is defined as hc/K_b.

The second source of radiation detected by the camera comes from reflections of radiation emitted by surrounding objects off the object being observed. The detected radiative power flux that is attributed to reflections is given in Equation 6.

$$Q_{refl} = f(\lambda_1 T_{surr}, \lambda_2 T_{surr})(1-\varepsilon)\tau\sigma T_{surr}^4 \quad (6)$$

Notice that compared to Equation 3 T_{obj} has been changed to the surrounding objects' temperature T_{surr} as they will emit the source of reflected radiation. Furthermore, the reflectivity of the object is used rather than the emissivity. An object's reflectivity is defined as (1-ε).

The last source of radiation the microbolometers detect is from the atmosphere. Due to Kirchhoff's Law (emissivity is equal to absorptivity), since the atmosphere absorbs some amount of the incoming radiation, it is also expected to emit some amount of radiation [7]. The detected radiative power flux attributed to atmospheric emissions is defined in Equation 7.

$$Q_{atm} = f(\lambda_1 T_{atm}, \lambda_2 T_{atm})(1-\tau)\sigma T_{atm}^4 \quad (7)$$

The detected radiative power flux incident with a microbolometer can now be written as seen in Equation 8.

$$Q_{tot} = Q_{obj} + Q_{surr} + Q_{atm} \quad (8)$$

Substituting the curve fitted relationship between measured microbolometer resistance and detected radiant power flux for each pixel provides a correlation between a pixel's measured resistance and the temperature of the imaged object. The equation is then rearranged to solve for T_{obj}, thus the relationship between the object's temperature and the measured resistance of the microbolometer is now formulated in Equation 9.

$$T_{obj} = \sqrt[4]{\frac{f_i(R) - Q_{surr} - Q_{atm}}{f(\lambda_1 T_{obj}, \lambda_2 T_{obj})\varepsilon\tau\sigma}} \quad (9)$$

To solve T_{obj}, the user must provide the emissivity of the object, the transmittance of the atmosphere, and the surrounding and atmospheric temperatures (which are often assumed as the same). Note that T_{obj} appears on both sides of the equation, thus T_{obj} must be solved iteratively. This solves the temperature for a single pixel in a thermal camera.

1.2 Emissivity and Thermal Imaging

As stated in the previous section, a user must supply a thermal camera's software with the emissivity of the object(s) being imaged, the atmospheric transmittance, surrounding temperatures, and atmospheric temperatures. In general, one can reasonably assume the atmospheric transmittance, surrounding temperatures, and atmospheric temperatures is the same for all pixels

in the microbolometer array. On the contrary, if the imaged scene contains various objects of different emissivities, then the pixels corresponding to the measurement of each materials' temperature must use the correct emissivity for that material to obtain accurate measurements. This has long presented a problem in thermal imaging. Figure 4 shows how failure to compensate for the discrepancy can result in drastically inaccurate measurements. The room temperature infrared image's Vanderbilt logo, Figure 4B, appears cold with respect to the non-painted portion of the metal slab. The non-painted metal is reflecting much of the camera user's body heat, whereas the painted portion does not reflect a significant amount of heat. In the high temperature scene, Figure 4C, the radiant power emitted from the high emissivity paint is greater than the sum of the reflected body heat and radiant power emitted from the low emissivity metal surface causing the logo to appear warmer than the non-painted portion.

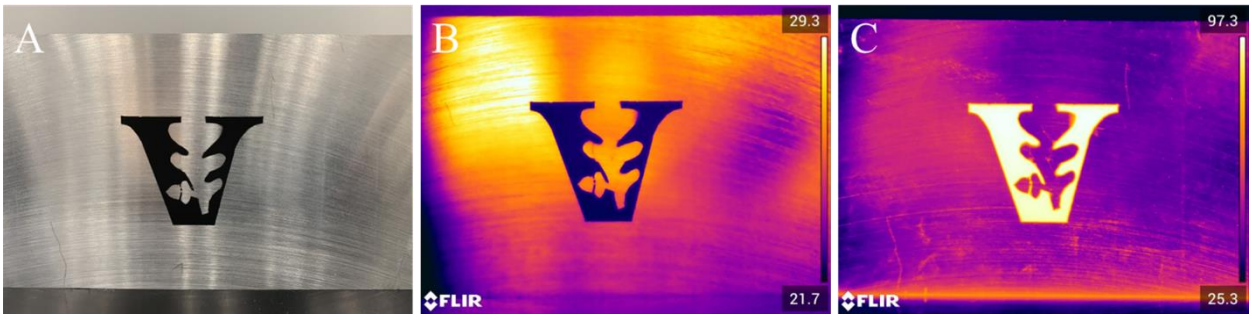

Figure 4: A metal slab (low emissivity) with a Vanderbilt logo designed using black paint (high emissivity) imaged in (**A**) the visible spectrum and the infrared spectrum with the metal slab at (**B**) room temperature and (**C**) after being warmed in an oven.

There are currently two predominant solutions to the emissivity problem in thermal imaging. The first consists of putting a small piece of tape or applying some paint of known

emissivity onto the observed objects and performing the temperature calculation using the emissivity of the tape or paint. The measured temperature of the object at the location of the paint or tape is assumed relatively constant throughout the body of the object [14]. This is not always an accurate assumption, nor is it always viable to put tape or paint on all objects of measurement such as in military surveillance applications.

The second solution utilizes defining regions of interest (ROI) around each unique material within the thermal imaging software and providing the materials' emissivities for each ROI [15]. This solution requires extensive interaction by the thermal camera's user and is therefore a cumbersome approach to the problem. This renders the solution unfeasible for application spaces that require near instantaneous measurements on thermal video feed such as autonomous driving. Furthermore, hasty manual creation of an ROI may result in imperfect boundaries between objects of differing emissivities causing inaccurate measurements along the edges of an ROI.

1.3 Deep Learning for Semantically Segmented Material Classification

This section will discuss how deep learning semantic segmentation provides a solution to the emissivity problem in thermal imaging presented in the previous section. First, we will lay the groundwork for a basic understanding of deep learning to better grasp how it applies to the problem. Then, we will discuss two generalized architectures of deep learning models that perform semantic segmentation classification. Finally, we will address why deep learning techniques provide the optimal solution to the emissivity problem.

1.3.1 Basics of deep learning

Deep learning is a subset of machine learning and artificial intelligence that utilizes artificial neural networks (ANNs) with many layers to learn and extract meaningful features from data to perform some task [16]. These methodologies have been used in various applications such as computer vision, speech recognition, language translation, drug design, and many more [17-20]. This subsection will focus on the convolutional neural networks (CNNs) subset of deep learning as it applies best to the emissivity problem and is the means of performing semantic segmentation. CNNs excel at computer vision tasks because they can contextualize patterns and shapes within images to make a well-informed prediction of what the CNN is observing [21]. These networks tend to consist of several types of layers including fully connected, convolutional, pooling, and occasionally upsampling layers.

Fully connected layers (sometimes called dense layers) serve as the basis for conventional ANNs. Many CNNs apply one or more fully connected layers to flattened data at the end of the neural network. These layers can form increasingly abstract representations of data as more layers are included. However, neural networks consisting of only fully connected layers often suffer from overfitting and poor generalization in computer vision tasks [22]. Each layer is comprised of many neurons or nodes and each neuron connects to all the neurons of the previous layer as depicted in Figure 5 [23]. The connections are formed via a dot product of a neuron's weights and the neurons' outputs from the previous layer. A neuron applies an activation function such as ReLU, seen in Equation 10, to the result of the dot product before passing its value to the following layer [24].

$$f(x) = max\{0, x\} \tag{10}$$

Training the neural network updates the weights associated with each neuron starting with the layers at the end of the network in a process called backpropagation. The weights are updated according to stochastic gradient descent of a given loss function.

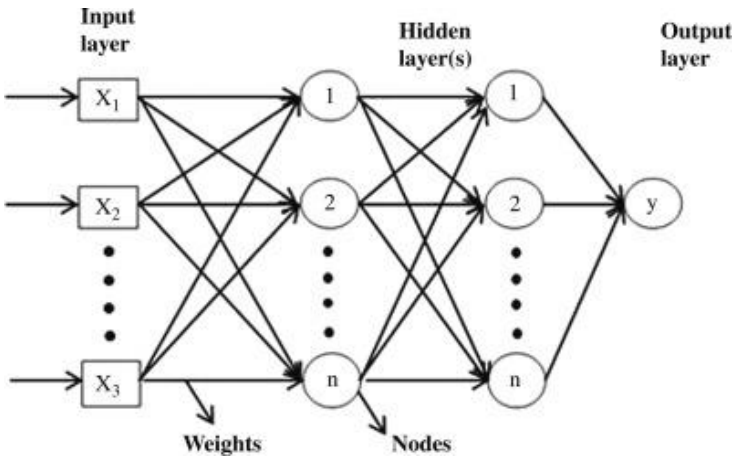

Figure 5: An ANN consisting of only fully connected layers.

Convolutional layers stride several convolutional kernels called filters across data output from previous layers. Each filter contains a set of trainable weights. In a single step of the stride, the weights are multiplied in an element-wise fashion with the values of the input, and the sum of this multiplication is the output. The final output of the convolution is a matrix of all strides. Each filter produces its own matrix in the output. This is illustrated in Figure 6 [25]. The geometry of the kernel is what enables CNNs to identify shapes and patterns exceedingly well. As a convolutional kernel undergoes training, its weights change in a manner that causes the kernel to output a high value when it detects a pattern that matches the pattern it is searching for and a low value when it does not. As a result, each filter becomes specialized in detecting a specific pattern. For instance, one filter may specialize in identification of curved edges while another specializes

in identification of straight edges [26]. Since the filters stride over the entire input matrix, the identification is location-invariant which significantly helps the generalization of these networks.

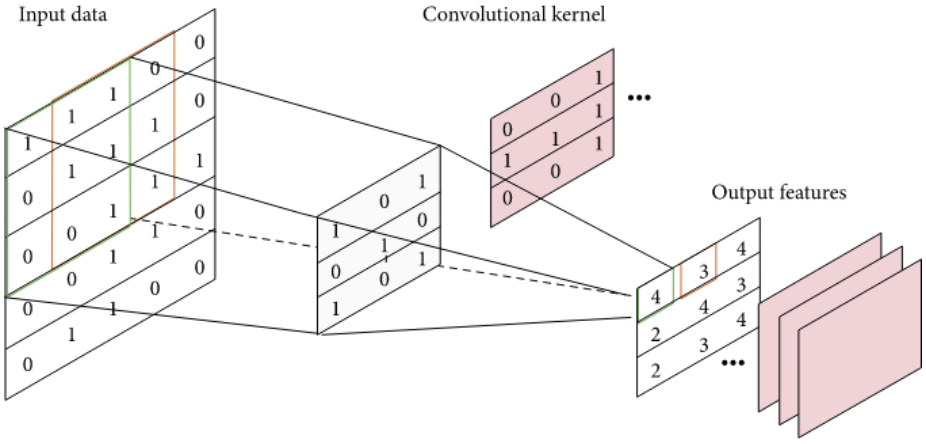

Figure 6: Convolutional kernels applied to input data and the resulting output data.

Pooling layers help a CNN to discard unneeded information via dimensionality reduction of the input matrix while conserving useful information. There are many types of pooling, but the most used is max pooling. The max pooling kernel is often a 2x2 kernel that strides over the input matrix without overlap. For a single stride, its output is the maximum value of the area contained by the kernel, and the remaining values are discarded. A schematic of this process is seen in Figure 7 [27]. Since a max pooling kernel simply passes along the maximum value and discards the rest, it does not have any trainable parameters. Max pooling is a sensible choice of pooling given the convolutions that it follows output a higher value for applicable patterns and low values for nonapplicable patterns, thus only the most valuable information is passed into the output [28].

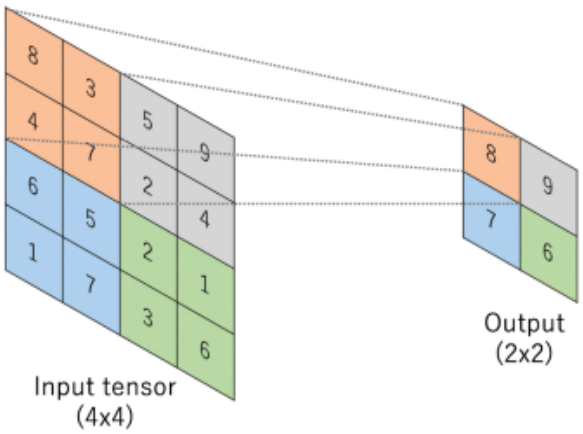

Figure 7: Input and output tensor before and after applying max pooling.

Upsampling is used in the special case of encoder-decoder type CNNs. This type of CNN will be expanded upon in the following section. Dimensionally, an upsampling layer performs the opposite operation as a pooling layer. Upsampling takes a single value and places it within a 2x2 area. The remaining three values are either the same value as the original single value, or they are zeros. In the first case, no parameters are learned for the layer. In the second case, the index of the original value is either always assigned to the top-left location in the new 2x2 area, or the index of its location is a trained parameter. The latter is shown in Figure 8 [29]. This layer is added when it is necessary to add resolution to the previous layer's output.

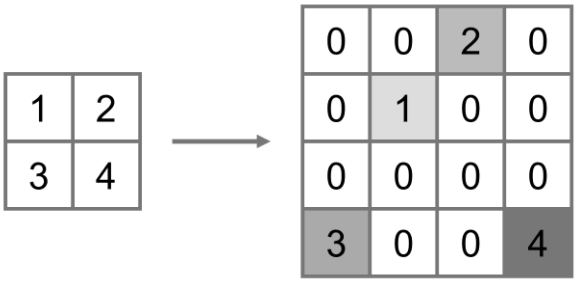

Figure 8: Upsampling with trained index placement.

The final layer in a multiclass ANN or CNN classifier has as many neurons along the classification dimension as the amount of target classes. Instead of ReLU, the activation function used is softmax. The softmax activation function normalizes the outputs into a probability distribution of predicted classes. Thus, the class that the neural network believes the data in question most likely falls into will have the highest value. Because it is a probability distribution, the sum of the softmax for all possible classes must equal one. The softmax function is given in Equation 11 [30].

$$\sigma(\vec{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \tag{11}$$

The equation normalizes the input vector from the final layer of the model, $\vec{z}$, such that the model predicts class *i* to have the probability $\sigma(\vec{z})_i$ of being the correct class out of K classes.

1.3.2 Semantic segmentation model architectures

Semantic segmentation entails creating a pixelwise classification map of an entire image. An example of an input image and its semantically segmented output is seen in Figure 9 [31]. There are several types of deep learning architectures which perform semantic segmentation. One of these models is the sliding-window (or patch-based) CNN and another is the U-net CNN [32, 33].

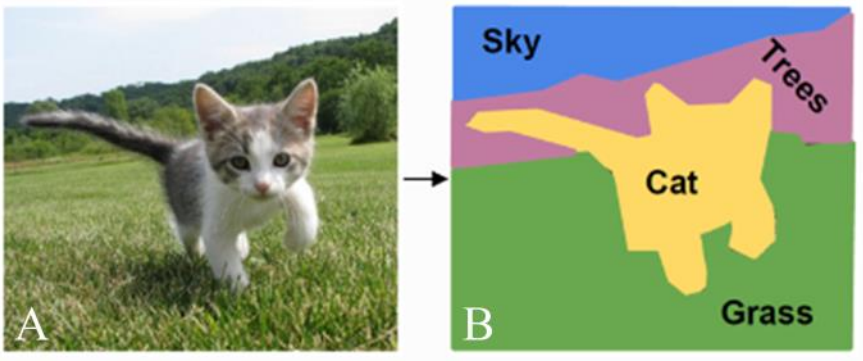

Figure 9: (**A**) Image of a cat in a field and (**B**) the corresponding semantic segmentation of the image.

The sliding-window CNN inputs a small patch of pixels from a larger image and classifies the central pixel of the patch. As the name suggests, preprocessing is performed to extract many overlapping patches of pixels as if a small window was slid over the image and at each step the contents within the window were classified. A CNN consists of one or more convolutional layers, followed by flattening of the data, and then one or more fully connected layers. The final layer is a fully connected layer using the softmax activation function to perform the classification [34]. A schematic of the architecture is given in Figure 10. This CNN differs from a standard CNN architecture as it does not apply pooling because the input patch is too small to lose any spatial data. A drawback of the patch-based CNN approach is that the small input restricts the model from contextualizing information from the greater image which may help with the classification task. Furthermore, the patch-based CNN's output loses some resolution from the input due to not evaluating the edges of the input. Edges are not segmented because some pixels' corresponding patch falls outside of the image. Padding the edges of the image with black pixels can recover the resolution, but this can cause poor classification accuracy where the padding is necessary.

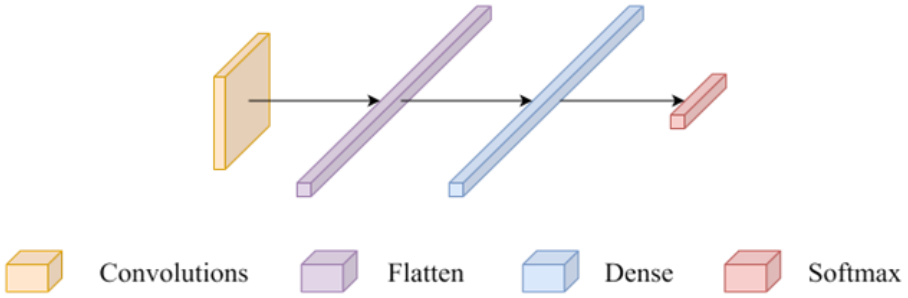

Figure 10: Generalized CNN architecture of the patch-based approach to semantic segmentation.

The U-net CNN is a fully convolutional approach to semantic segmentation tasks. The architecture follows an encoder-decoder type structure. The encoding portion takes an entire image as input data and compresses it into a smaller dimensionality via convolutions and pooling to discard excess information in the image. A decoder portion follows, which projects the compressed data into the classification space using convolutions and upsampling. Skip connections are used throughout the model to symmetrically pass data from the encoding portion to the decoding portion [33]. The architecture is visualized in Figure 11. The skip connections are used to prevent the vanishing gradient problem in neural networks with many layers. The vanishing gradient problem prevents layers early in the network from optimally learning the proper weights. This is because changes in weights have a diminishing effect on the output of the neural network the further the weights are from the classification layer. Incorporating skip connections gives weights that appear early in the neural network a more direct effect on the output of the neural network. This enables the design of a deeper neural network that can create more complex abstractions of the input data than would otherwise be possible without skip connections [35]. Since the U-net model inputs an entire image at a time, it can use the contextual information of the entire image for classification

that the patch-based models are incapable of utilizing. Performing classification with U-net models is generally quicker than the patch-based model because they do not require overlapping patches from the image. This means that a single pixel is only processed by the model once for the semantic segmentation task, thus less overall computation is required. As a result, U-net models are better fit for applications where the classification map must be generated quickly such as in autonomous vehicles.

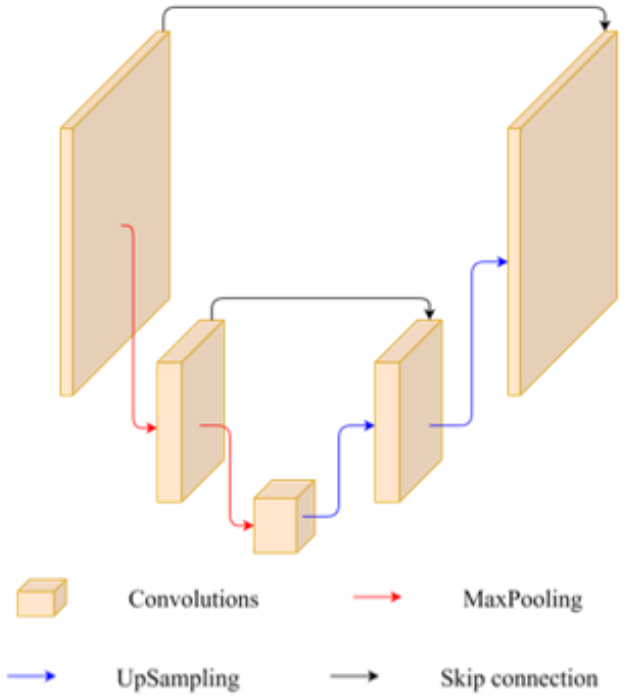

Figure 11: Generalized CNN architecture of the U-net approach to semantic segmentation.

1.3.3 Solving the emissivity problem with deep learning

This work proposes using the deep learning techniques previously discussed to create a semantic segmentation map of the materials within a thermal image. Once the material imaged by

each pixel is identified, the emissivity of said material can be looked up in a database provided to the thermal imaging software. At this point, the temperature of each pixel is recalculated using the correct emissivity. This circumnavigates the need to apply additional materials to the objects of the scene or to draw ROIs around each object by hand and then manually specify the correct emissivity. Effectively, the deep learning model creates the necessary ROIs, and the software fills in the applicable emissivity to each ROI. This expedites the process required to obtain accurate temperature measurements with little to no additional effort from the thermal camera's user. If a model can achieve very high accuracy for the material classification semantic segmentation map, then the results may produce even more precise results than manual creation of the ROIs.

1.4 Multispectral Imaging to Enhance Material Classification

Thermal images provide no information on the unique spectral surface radiative characteristics of a material. A thermal camera that captures radiation in the 7.5 μm to 14 μm spectral window does not have the ability to discern the precise spectral origin of the electromagnetic radiation. This inability results in the loss of material specific spectral information that the deep learning algorithm could leverage to perform the semantic segmentation with higher accuracy [36]. A thermal image is akin to a grayscale image as opposed to an RGB image in the visible spectrum (0.4 μm – 0.7μm) because spectral characteristics govern the color of materials. For example, grass appears green due to chlorophyll's high scattering rate of mid-wavelength visible light (green) as opposed to its low scattering rate of high and low wavelength visible light (blue and red) [37], and this feature helps observers identify the material.

Further increasing the amount of spectral information in a thermal image, via hyperspectral imaging (HSI) or multispectral imaging (MSI), available to the deep learning model improves material classification accuracy [38]. Although dense spectral information is not inherently captured by a thermal camera, applying spectral filters to a thermal camera enables the extraction of spectral features in the FIR spectrum. A wide range of applications within geological sciences, military, and medical fields have taken a similar approach by incorporating HSI or MSI data to improve models' semantic segmentation performance for images in the visible spectrum [39-41].

In the FIR range, the radiosity (amount of radiation leaving a surface) is a combination of spontaneous emission and reflection. As a result, HSI and MSI in the FIR range provides information about both emissive and reflective features of a material in the thermal camera's spectral range, unlike visible images which are solely reflection [42]. However, most HSI and MSI semantic segmentation work is performed in the visible region. Specifically, three visible region datasets are predominantly used in the development of HSI segmentation models: Indian Pines, Salinas Scene, and University of Pavia datasets [43-45]. These datasets consist of airborne or satellite HSI of farmland and cityscape containing between 105 and 220 bands in the visible region. While most HSI semantic segmentation models are designed for visible region data, the core architecture of the models can be applied to semantic segmentation of FIR MSI given that modifications are made to accommodate for the number of bands in the FIR MSI dataset.

Objects in HSI data have been classified by employing several CNN architectures to semantically segment image scenes, such as the 1D-convolutional neural network (1D-CNN), which identifies a material's spectral features captured in a single spatial location [46]. The 2D-CNN, which was the type of CNN described in the previous section, limits the kernel to learning spatial features and treats the many HSI bands as color channels. This prevents the convolutional

kernel from explicitly learning spectral features [47]. The 3D-CNN, which enables spectral-spatial feature detecting kernels, has shown that learning the features across the spectral-spatial domain enables better classification performance than learning only a spectral or spatial component of the HSI data-cube [48]. These three types of CNNs are pertinent as all three of these feature types (spectral, spatial, and spectral-spatial) are observed in FIR MSI data.

The Hybrid-Spectral-Net (HybridSN) incorporates principal-component-analysis (PCA) to disregard bands with little information and then feeds the data through a series of 3D convolutions to identify spectral-spatial features. The data are then reshaped and passed through 2D convolutions to further identify spatial features. The HybridSN demonstrates that independently learning spectral-spatial and spatial features both reduces computational time for training and increases classification performance [49]. All the models mentioned previously employ single patch-based segmentation. The spectral-spatial multi-scale feature fusion network (SMFFNet) supports a spatially small spectral-spatial patch input containing all bands and a spatially large spatial patch input containing few bands. The two inputs are put through a series of independent and parallel 3D and 2D convolutions before the data streams are fused. This separates the spectral-spatial and spatial feature detection into their own sections of the networks. The model has achieved nearly perfect metrics across the standard visible region HSI datasets [50]. The U-Net model has also been tested on HSI datasets, but due to the identical input and output dimensionality nature of the U-Net the spectral bands are input as color channels [51]. Much like the 2D-CNN, this restricts the model to only detecting spatial features.

CHAPTER II

METHODS

2.1 Experimental Methods

We created a dataset of MSI thermal images using the FLIR A655sc High-Resolution LWIR science-grade infrared camera in conjunction with FLIR ResearchIR software. All thermal images were captured in 3 different spectral ranges. The first spectral range covers 7.5 μm to 14 μm, the spectral range inherent to the thermal camera. The two other spectral ranges cover 7.5 μm to 9 μm and 8.5 μm to 14 μm by inserting Andover short-pass and long-pass edge filters into the thermal camera, respectively. Each thermal image consists of eight unique material blocks (aluminum, acrylic, bakelite, cork, ethylene-vinyl acetate [EVA], granite, silicone, and maple) placed on a ceramic sample holder. The sample holder and materials were placed in an oven, heated to temperatures ranging from 30 ^{o}C to 55 ^{o}C with measurements taken every 5 ^{o}C. The temperature of the samples was monitored with a thermocouple. The room temperature was recorded as 20.8 ^{o}C. At each temperature, seven permutations of a two by four stacking of the blocks were constructed. For each permutation, one image was collected for each of the three filters, giving a total of 21 images per temperature. With six distinct temperatures, a total of 42 scenes were imaged yielding 126 thermal images (7·3·6 = 126). After changing the oven temperature, the material blocks were allowed to equilibrate for 20 minutes at the new temperature. Opening the oven and handling the blocks to obtain permutations disrupts the material blocks' temperature. Thus, between changing the permutation of the blocks and imaging the scene at different spectra, the blocks were allowed to equilibrate for 10 and 5 minutes at the target temperature, respectively. An

ROI containing only the ceramic background and material blocks was applied in the ResearchIR interface to avoid imaging the interior walls, racks, and heating elements within the oven. The restricted ROI yielded thermal images with resolution of 294x181 pixels for all images collected. The ground truth for each scene was created in ImageJ by outlining each block and assigning it a color corresponding to its material [52]. A single permutation's set of MSI thermal images and the corresponding ground truth is shown in Figure 12.

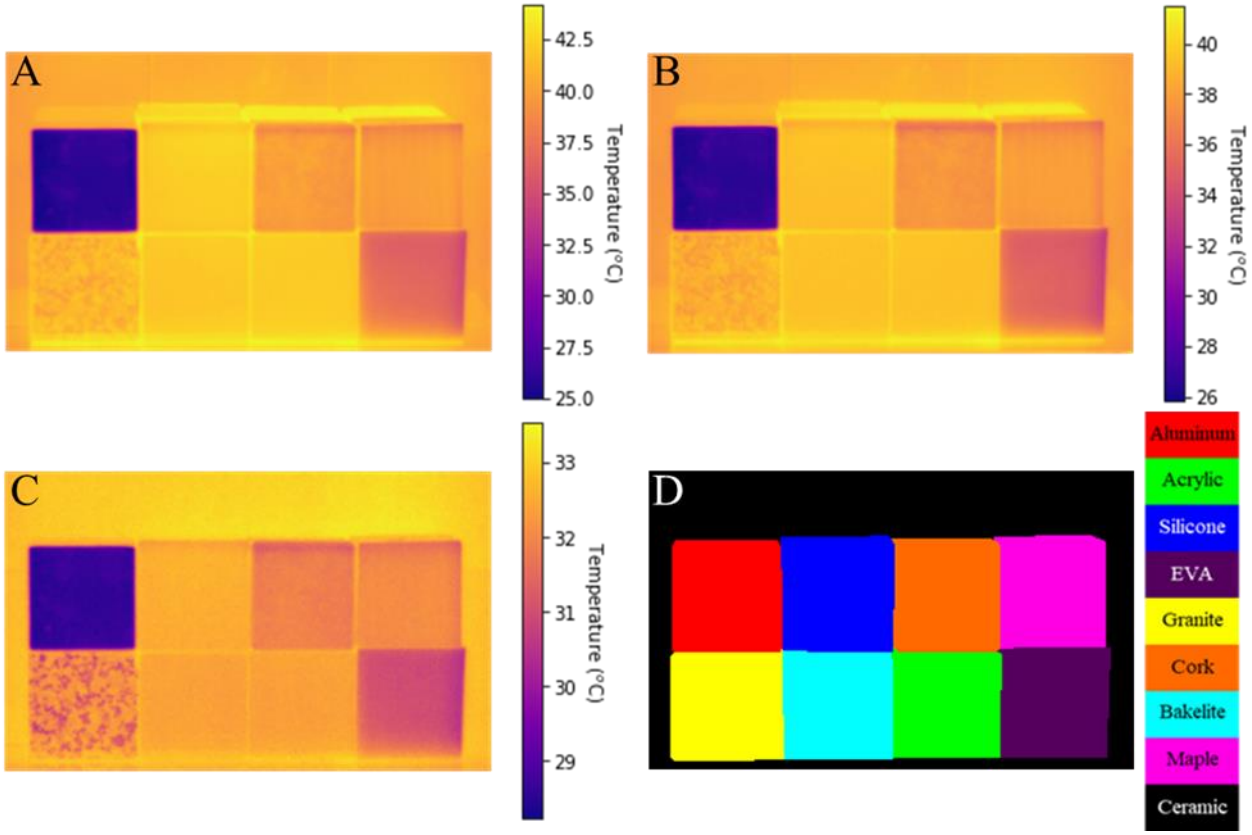

Figure 12: Thermal images of the test scene taken with the oven set to 50 °C for the (**A**) full spectrum, (**B**) long-pass, and (**C**) short-pass spectral ranges. **D** is the ground truth for the scene. (Colors ascribed to ground truth image do not correspond to the false color assignment in the thermal images; however, material locations do correspond.)

2.2 Computational Methods

2.2.1 Data preprocessing

Let the dimensionality of the dataset be represented by $\mathbf{I} \in \mathbb{R}^{T \times P \times M \times N \times B}$, where T is the number of different temperatures for which data were collected. P signifies the number of permutations collected at each temperature. M and N express the height and width, respectively, of pixels in the thermal images. B gives the number of unique spectral windows at which each scene is imaged. The data were split into three sets: 5 of the 7 permutations for training, 1 of the 7 for validation, and 1 of the 7 for testing. Each (M x N) thermal image was normalized such that the minimum and maximum temperatures were 0 and 1, respectively, in each scene with the remaining temperatures in the range [0,1]. For all data in **I**, preprocessing steps to realize eventual pixel-wise segmented classification was performed by independently extracting and vectorizing overlapping spectral-spatial data-cube patches and spatial data-square patches for each of the (M x N x B) scenes. Corresponding spectral-spatial data-cube and spatial data-square patches share the spatial location of their central pixels. Let each individual spectral-spatial patch's dimensionality be represented by $\mathbf{W} \in \mathbb{R}^{S1 \times S1 \times B}$, where S1 represents both the spatial height and width of the data-cube patch. Similarly, let the individual spatial patch's dimensionality be given by $\mathbf{U} \in \mathbb{R}^{S2 \times S2 \times 1}$, where S2 represents both the spatial height and width of the data-square patch. The unity value in the third dimension of **U** signifies that the patches were only extracted from the full spectrum thermal image for this input. **W** and **U** were extracted from **I** with a spatial stride of (1 x 1) and no padding along the outer edges of the normalized thermal images. The central spatial pixel of **W** and **U** is the pixel to be classified by the neural network such that the output space is defined by a vector with elements $\mathbf{O} \in \mathbb{R}^{1 \times 1 \times B}$ meaning all spectral bands within a single spatially defined set of pixels are classified together.

The ground truths have a dimensionality of $\mathbf{G} \in \mathbb{R}^{T \times P \times M \times N}$. For each (M x N) scene in the ground truth dataset, a (M – S2 + 1) x (N – S2 + 1) key for the materials' spatial locations are equated to the color in each pixel of the ground truth with a one hot encoding label (a standard method of labeling the outputs of neural networks). The reason for the loss of spatial dimension size in the key is due to not using padding when extracting **W** and **U**. S2 determines the size of the output since the spatial patch has a larger height and width than the spectral-spatial patch. The one hot encoding label key is vectorized such that it defines the class corresponding to each element of the output space.

2.2.2 Deep learning models and metrics

All neural networks were developed using the Keras functional API of Tensorflow2. The design of the primary neural network used for material classification was inspired by the SMFFNet model proposed for handling HSI data with several modifications to better fit to our dataset (M-SMFFNet). The model is built with two input layers, one that accepts all data for **I** cast into **W** and another that accepts all data for **I** cast into **U**. The input layer that **W** feeds into has a shape of (S1 x S1 x B x 1), where the 1 specifies that only one color channel is present in the data. This input is followed by a 3D convolution with a kernel size of (2 x 2 x 2) to detect the spatial-spectral features within the data. The input layer for **U** is shaped as (S2 x S2 x 1) since the input contains no spectral data. This input continues into two 2D convolutions with (3 x 3) kernel sizes. Following the convolutions of both inputs, the data is flattened and then fused using a concatenation layer. The concatenated data are subsequently passed through two dense layers containing 128 units. Finally, a softmax layer is applied to perform the classification. The softmax contains nine classes, one for each material block and an extra for the ceramic background. The neural network is

depicted in Figure 13. All convolution and dense layers utilize L2 regularization on the kernel and bias, both with coefficients of $3x10^{-3}$ to prevent exploding gradients. Additionally, all convolution and dense layers are batch normalized before applying a ReLU activation. The loss function was calculated via categorical cross-entropy. An ADAM optimizer was utilized with the initial learning rate set to $1x10^{-5}$. The model was allowed to train for a maximum of 1000 epochs, but early stopping was enforced upon diminishing returns on the validation set's categorical accuracy. The patience for improving validation categorical accuracy was set to 50 epochs. A callback was employed to restore the weights from the best performance on the validation set.

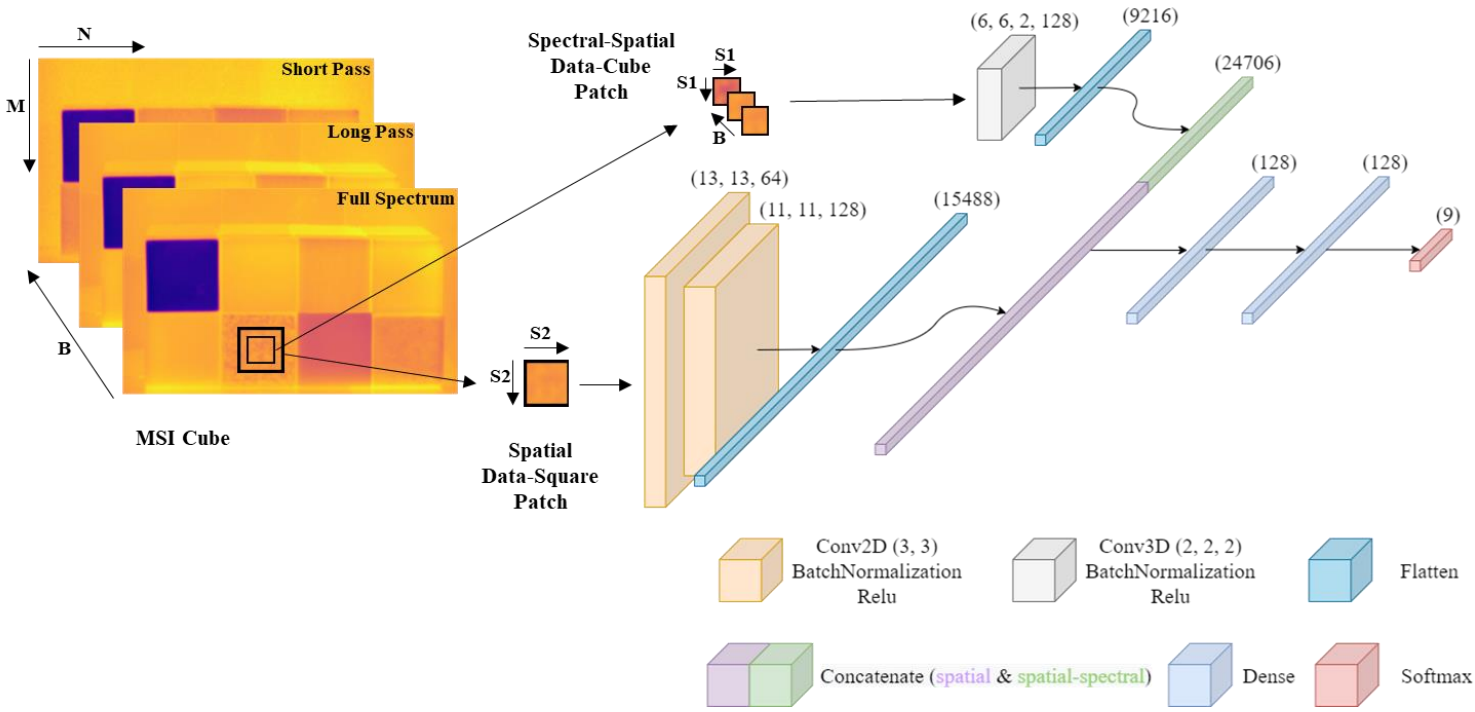

Figure 13: A detailed schematic of extracting spatial-spectral and spatial patches from the MSI cube captured in Figure 12, and the subsequent M-SMFFNet that classifies the central pixel.

Three other models were also trained and tested. The M-SMFFNet was compared with a modified HybridSN (M-HybridSN), a U-Net, and a custom 3D-2D U-Net developed for this work. The M-HybridSN takes its input from **W** and returns an output of **O**, except the transformation of **G** to **O** is defined by S1 rather than S2 because **U** is not included in the model**.** The U-Net and the 3D-2D U-Net both take in **I**, which has been reshaped to stack the temperature and permutation into the same dimension. They both output in the dimensionality of **G** with the temperature and permutation dimensions stacked in the same manner as the input. Detailed schematics of all three of these models can be found in the supplemental information.

All neural networks were evaluated against three metrics. The first is overall accuracy of the test scenes. This is computed by dividing the number of correctly predicted pixels by the total number of pixels. Overall accuracy tells how well the model segmented the entirety of the images. Second, we observe the average accuracy, which is solved via dividing the number of correct pixels for each material by the true number of pixels representing that material. These per material accuracies are then summed and divided by the total number of materials to yield the average accuracy. This metric better accounts for imbalances between the number of pixels representing each material in the data. Third, we evaluate Cohen's Kappa, which is given by subtracting the probability of the predictions agreeing with the true values (P_e) from the overall accuracy and then dividing this difference by 1-P_e. Cohen's Kappa assesses the model's performance against segmenting the model by chance. The metric also identifies models that perform exceptionally well in cases of imbalanced data.

2.2.3 Data postprocessing

After determining the material occupying each pixel in the testing set, images of the predictions are generated. The results are evaluated using the metrics described in the previous subsection, along with a confusion matrix. The semantic segmentation map of materials and their corresponding emissivities is then used to calculate the corrected temperature using Equation 9. Once all the temperatures are corrected, a new heatmap of each test scene is generated. Average corrected object temperatures are calculated by taking the mean corrected temperature of the pixels corresponding to the ground truth for each object.

CHAPTER III

RESULTS AND DISCUSSION

3.1 Material Classification

Each models' performance can be visualized against the ground truth in Figure 14. Doing so reveals that the U-Net architectures greatly reduce the salt and pepper classification noise seen in the patch-based architectures. This is the small, scattered patches of misclassified pixels seen in Figures 14C and 14E. However, this comes at the expense of edge detection. The U-Net architectures tend to have more difficulty distinguishing between similar materials when they are placed next to each other, resulting in the model producing poor segmentation boundaries as seen in Figures 3B and 3D between the acrylic, bakelite, and silicone, as well as the cork and maple. While the patch-based architectures manage to define the boundaries better, there is significant salt and pepper noise around the edges of each block.

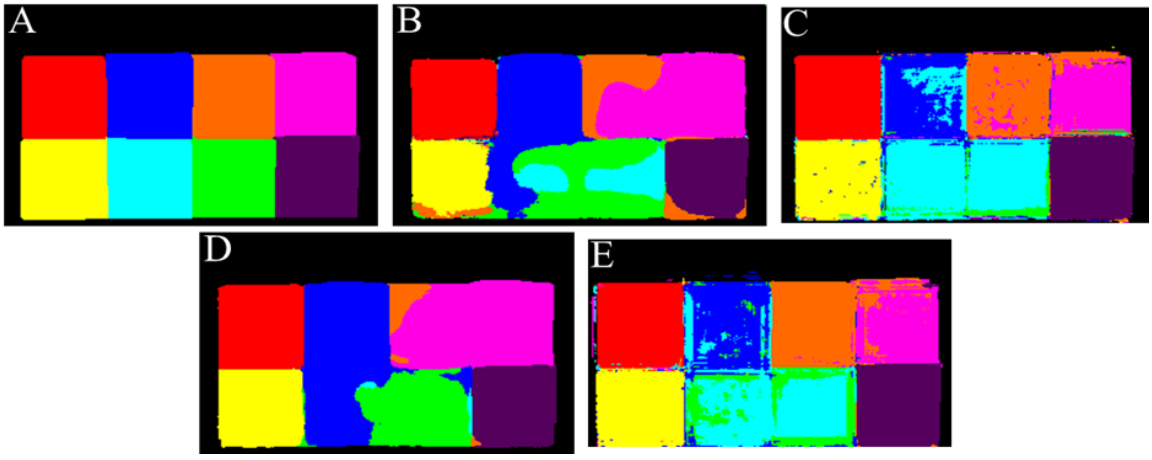

Figure 14: (**A**) Ground truth, (**B**) U-Net, (**C**) M-HybridSN, (**D**) 3D-2D U-Net, and (**E**) M-SMFFNet semantic segmentation material classification of the test scene at 50 ^{o}C.

The performance of each model evaluated against all the selected metrics is given in Table 1. The M-SMFFNet performed the best in most of the per material accuracy metrics, and every compiled metric (OA, AA, Kappa). The 3D-2D U-Net performed second best in two of the three compiled metrics but does worse than the M-HybridSN in average accuracy. This is due to the model overfitting to silicone and acrylic when attempting to predict bakelite. The poor performance on bakelite had a significant impact on the model's average accuracy. The same overfitting issue occurred with the classic U-Net as well. It should be noted that the traditional U-Net, the only model that does not consider spectral-spatial information, does significantly worse on granite than any other model. Looking back to Figure 12, granite has the most substantial visual difference between bands, thus suggesting that it has significant changes in its surface radiative properties depending on the spectral region observed. This explains why the U-Net model, which does not explicitly learn spectral features, is unable to classify granite as accurately as the other models.

Table 1: Each models' performance metrics for all test scenes. The result of the model with the best performance for each category is highlighted in blue.

Material	**Model**			
	U-Net	M-HybridSN	3D-2D U-Net	M-SMSFFNet
Acrylic	60.5	30.9	70.0	39.3
Aluminum	95.2	99.4	99.1	98.9
Bakelite	19.0	73.3	16.2	70.0
Ceramic	98.2	97.2	98.7	96.5
Cork	85.0	87.6	81.7	94.9
EVA	93.9	98.2	97.9	98.4
Granite	75.8	95.3	94.3	97.4
Maple	76.3	65.3	70.2	75.0
Silicone	64.6	54.2	70.0	67.8
OA	80.5	82.3	83.1	84.5
AA	74.3	78.0	77.6	82.1
Kappa (x100)	76.4	79.1	79.6	82.1

A confusion matrix for M-SMFFNet is given in Figure 15. It is normalized by rows such that each row sums to one and each cell gives the fraction of pixels truly belonging to the material given along the vertical axis classified as the material along the horizontal axis. A perfect model would yield ones along the diagonal with zeros in all off-diagonal cells. All test temperatures are evaluated in the confusion matrix.

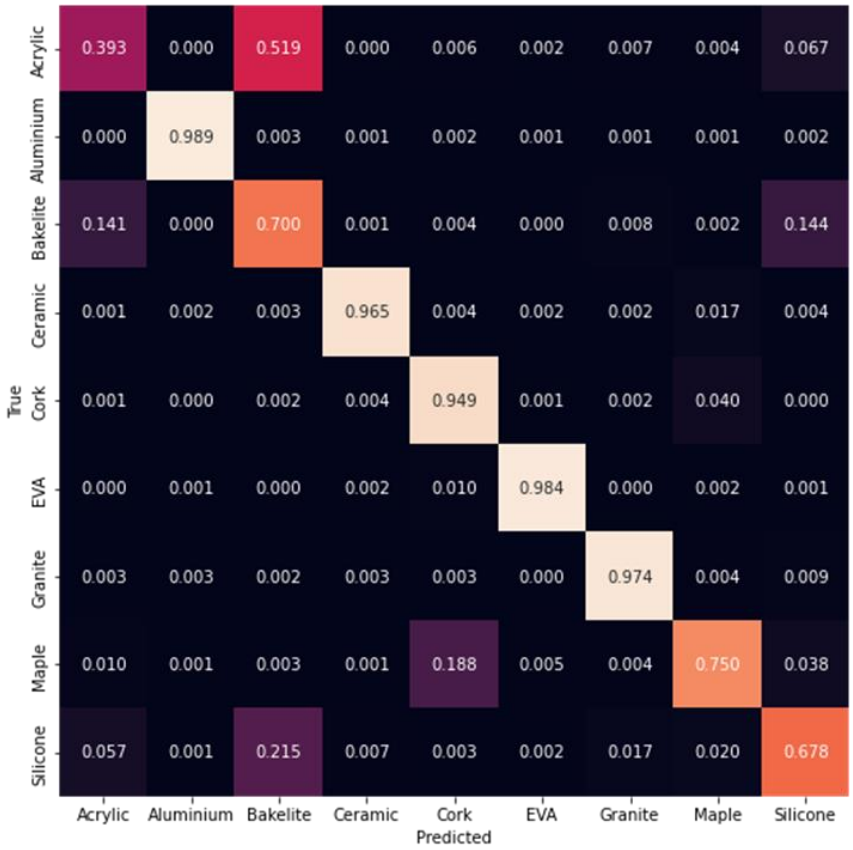

Figure 15: A confusion matrix of the predictions given by M-SMFFNet across all temperatures in the test set.

The confusion matrix further confirms the model's success in characterizing aluminum, ceramic, cork, EVA, and granite. However, the neural network fails to correctly identify acrylic, bakelite, silicone, and maple consistently. The confusion matrix, along with the prediction map in Figure 14E, reveals that the problem stems from the neural network having trouble distinguishing between

samples within each of the two groups of materials. The first group is acrylic, bakelite, and silicone, while the second group consists of cork and maple. Materials belonging to these two groups tend to have similar spectral features (emissivities as a function of irradiated wavelength) and few distinguishable spatial features (emissive patterns from a material's surface) among each other. Similar features are expected within the first group as these materials are all polymeric with homogenous smooth surfaces. The second group consists of wooden materials, hence the similar spectral features, yet the grains in the maple offer some spatial feature differences when compared to the unordered pattern of the cork surface. The model tends to predict materials falling into the first group as bakelite, and materials in the second group as cork, hence the increased material accuracies for bakelite and cork versus the other material(s) in their groups.

3.2 Temperature Correction

As seen in Figure 16A, temperatures along the edges of the aluminum block are corrected to an unreasonably high temperature. This is caused by the reflectance term within the correction calculation. T_{surr} is assumed to be the temperature of the room exterior to the oven; however, the edges of the aluminum are slightly curved and reflect heat from the interior walls and heating elements of the oven. Since the interior of the oven is much hotter than room temperature, which is not accounted for in the calculation, most of the reflected heat manifests itself within the emission term. As a result, the aluminum's edge temperature is perceived to emit a significant amount of energy, thus resulting in an overestimation of the corrected temperature. Another heatmap of the scene was generated with the maximum colormap value set to aluminum's average temperature. This allows the other materials' results to be visually analyzed as well. Comparing Figures 12A and 16B shows that the corrected temperatures have a much narrower distribution

than the uncorrected temperatures, except for the edges of the aluminum block. A narrower distribution is to be expected as the materials should all be at the same temperature having equilibrated in the same oven. A true thermal image would be uniform and show no detail.

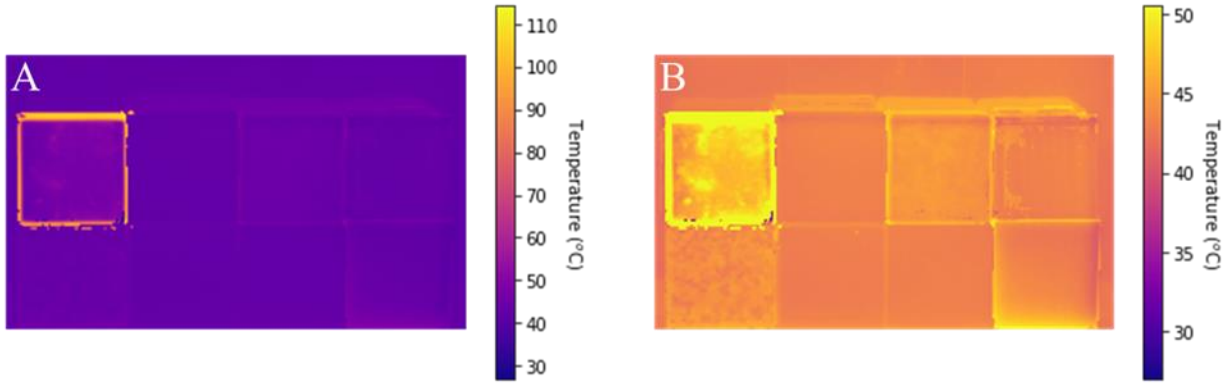

Figure 16: (**A**) The resulting temperature correction for the 50 ºC test scene, and (**B**) the same data with the heatmap capped to the average temperature of aluminum.

The result is further supported by the average material temperatures in Figure 17. The average temperature of aluminum is typically higher than the others, again due to the edges of the block reflecting heat from the oven. Despite the neural nets' inability to distinguish materials within the two groups mentioned in the previous section, the temperature correction still obtains favorable results. Since the spectral characteristics of the materials in each group are similar, their emissivities are approximately the same. Thus, the temperature correction comes out to the same value regardless of the predicted material assuming the prediction falls within the group that each material resides. While the M-MSFFNet was able to accurately predict the pixels containing granite, the composite material has a nonuniform emissivity across its surface. A single emissivity is used for the entirety of the granite surface in the temperature correction. This results in the

average temperature of the granite approaching the expected value, yet the surface maintains a significant temperature fluctuation.

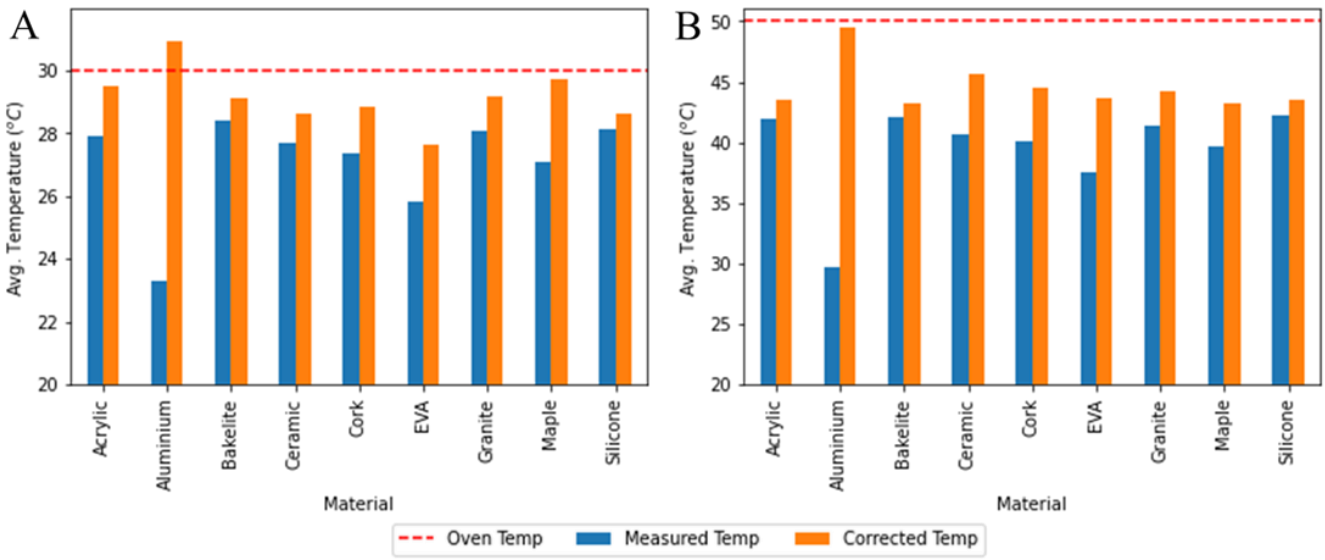

Figure 17: Average temperature of each object's surface before and after the correction for test scenes (**A**) T_{oven} = 30 °C and (**B**) T_{oven} = 50 °C.

The root mean square error between average object surface temperature and the oven temperature for both measured and corrected cases across all temperatures is shown in Figure 18. While it is not expected that the root mean square error (RMSE) be exactly zero, due to the cooling of the blocks upon opening the oven, we do expect the RMSE to approach zero for all materials. Approximating the time constant for cooling from both radiative and convective losses suggests that a negligible amount of cooling occurs before an image is taken, therefore the assumption that the RMSE should have a value close to zero is affirmed. Applying the temperature correction yields a decrease in the spread of the RMSE from 9.8 °C to 2.8 °C. The average RMSE across all materials decreases from 8.2 °C to 4.3°C.

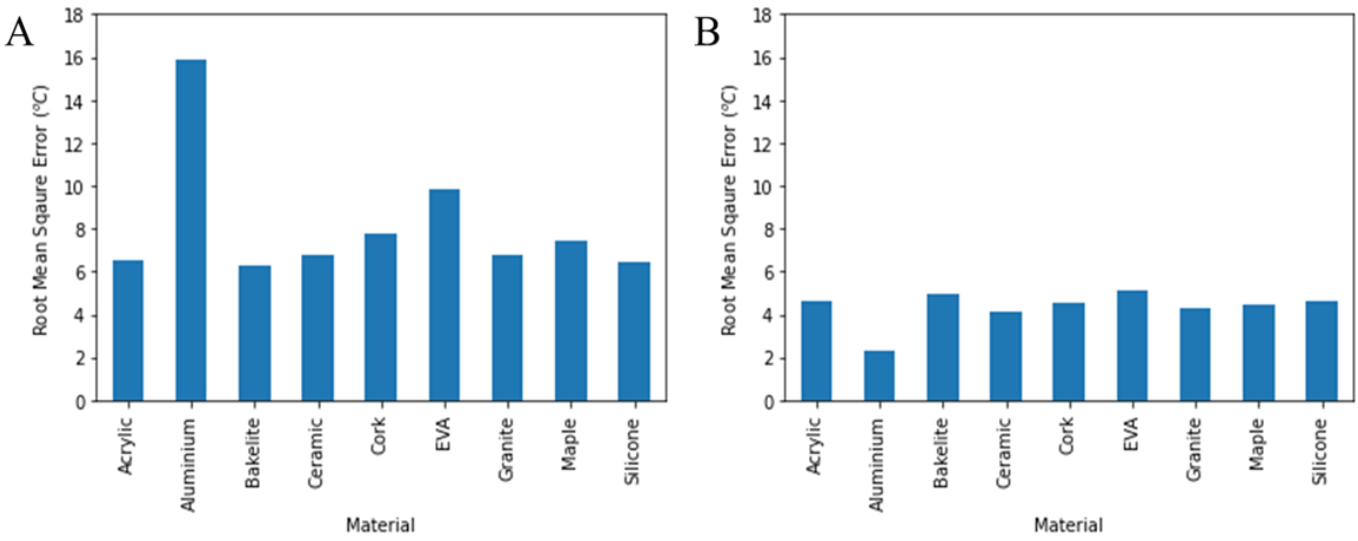

Figure 18: Root mean square error of average surface temperature for each material across all test scene temperatures between the (**A**) measured temperature and (**B**) corrected temperature against the expected temperature (T_{oven}).

CHAPTER IV

CONCLUSIONS

Material classification by applying a CNN to multispectral thermal images is accurate given that the materials have unique spectral and spatial features. The M-SMFFNet provides the best semantic segmentation results out of the models tested. The 3D-2D U-Net performs nearly as well without the salt and pepper noise inherent in patch-based segmentation schemes. All tested neural networks generally fail when the spectral features are similar and tend to overfit to a particular material within a group of similar materials (such as polymeric plastics or wood-based materials). Despite the neural networks' failure to discern these materials, the similarity between the spectral emissivities enables an accurate temperature correction. The temperature correction fails when calculating the temperature of highly reflective materials if the surrounding temperatures are not uniform. Composite materials with nonuniform emissivities enable high classification accuracies due to significant spatial features, yet the temperature calculation fails to account for the nonuniform emissivity across the surface. Temperature correction methodologies described in this work are best applied to materials with uniform surfaces.

CHAPTER V

FUTURE WORK

5.1 Incorporating the Visible Spectrum

By implementing RGB images into the dataset, the deep learning models will have more spectral information that they can use to help further improve the material classification performance. We have already begun this process by designing and 3D printing an iPhone attachment for the thermal camera. The attachment is seen in Figure 19. It was designed to align the point of views of the iPhone's camera and the thermal camera as closely as possible. Feature matching algorithms such as those provided by OpenCV may be used to perform the remaining image alignment necessary [53].

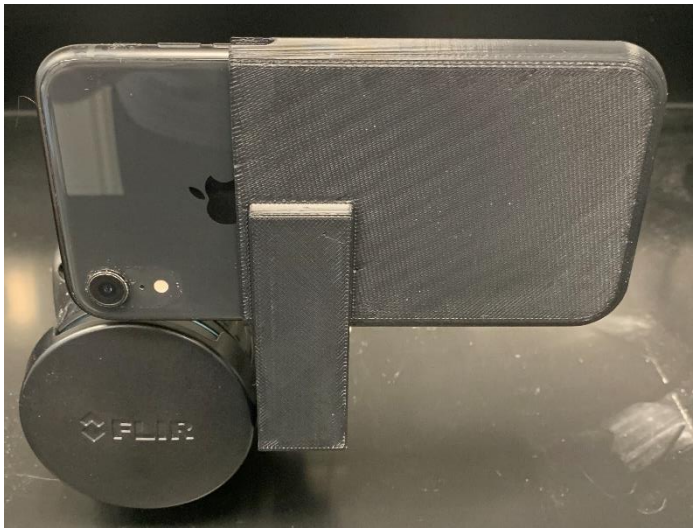

Figure 19: 3D printed FLIR thermal camera attachment for iPhone XR.

5.2 Transfer Learning

Visual tasks are interconnected and can benefit from sharing information between each other [54]. For instance, when somebody looks at a chair their brain may examine and utilize many features about the chair paired with some amount of prior knowledge to conclude that they are in fact observing a chair. These features may include the curvature of the seat, the color of the wood, the shading of the shadow cast below the chair, the shape that the edges of the chair form, etc. The taskonomy team at Stanford and UC Berkeley have shown that the identification of surface normals, curvature, and edge detection all play vital roles in the computer vision task of semantic segmentation [55]. Applying deep learning models which perform these specific tasks and incorporating the outputs into the semantic segmentation deep learning models may help the classifier generalize and have a more holistic understanding of the objects they are trying to semantically segment, thus improving the performance of the models.

SUPPLEMENTAL INFORMATION

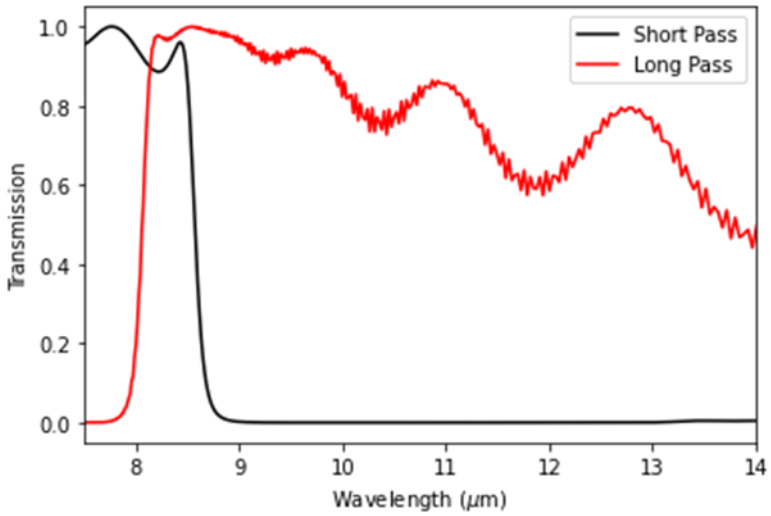

Figure S.I.1: Transmission curves of the long-pass and short-pass filters.

Table S.I.1: Emissivity values used to calculate the corrected temperatures [56-58].

Material	Emissivity
Acrylic	0.95
Aluminum	0.18
Bakelite	0.95
Ceramic	0.90
Cork	0.78
EVA	0.69
Granite	0.86
Maple	0.84
Silicone	0.95

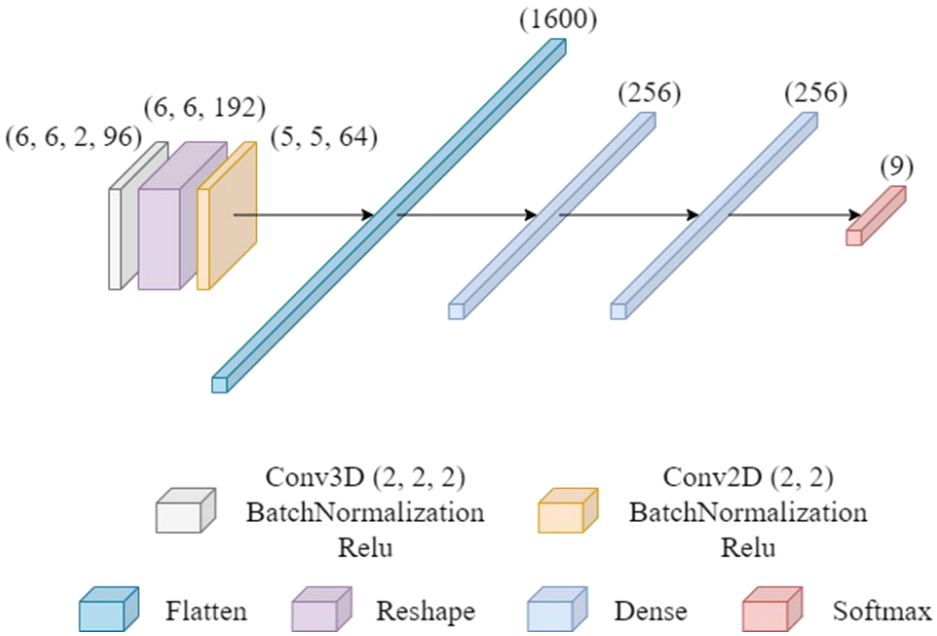

Figure S.I.2: Schematic of the architecture used for the modified-HybridSN model.

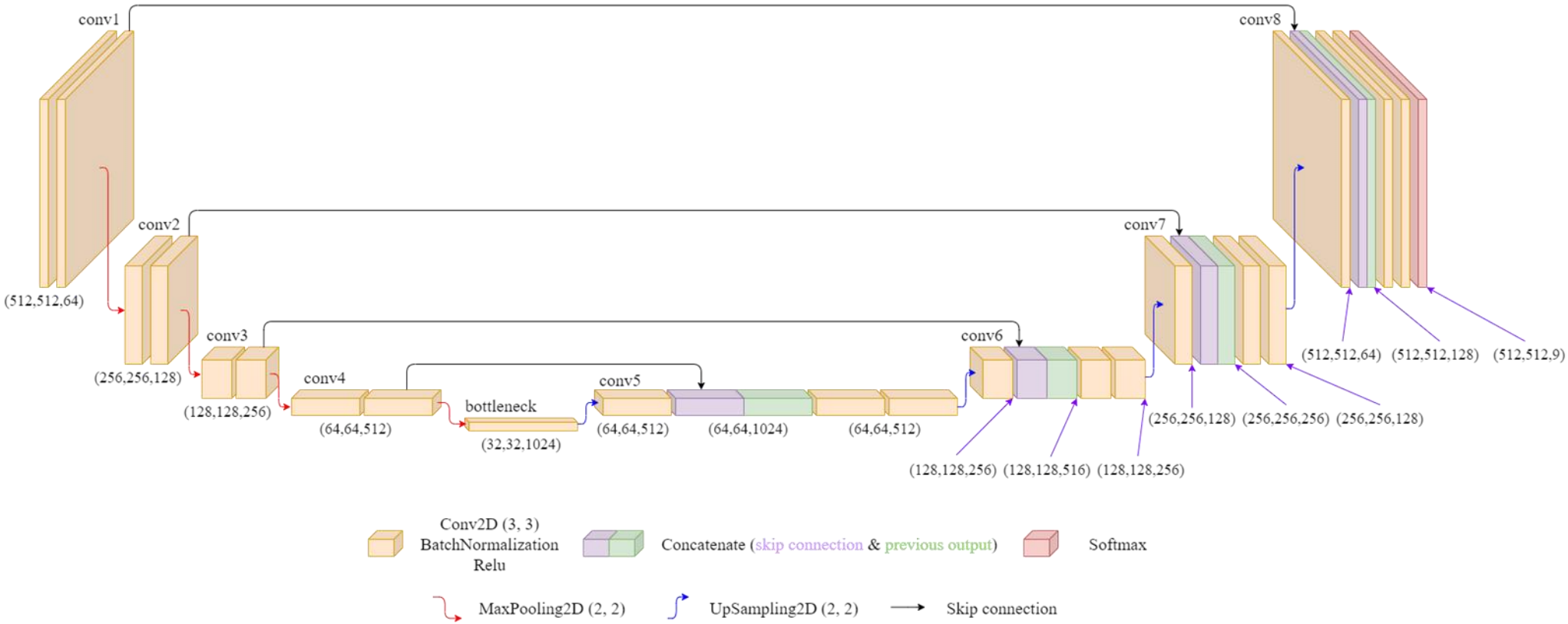

Figure S.I.3: Schematic of the architecture used for the 2D U-Net model.

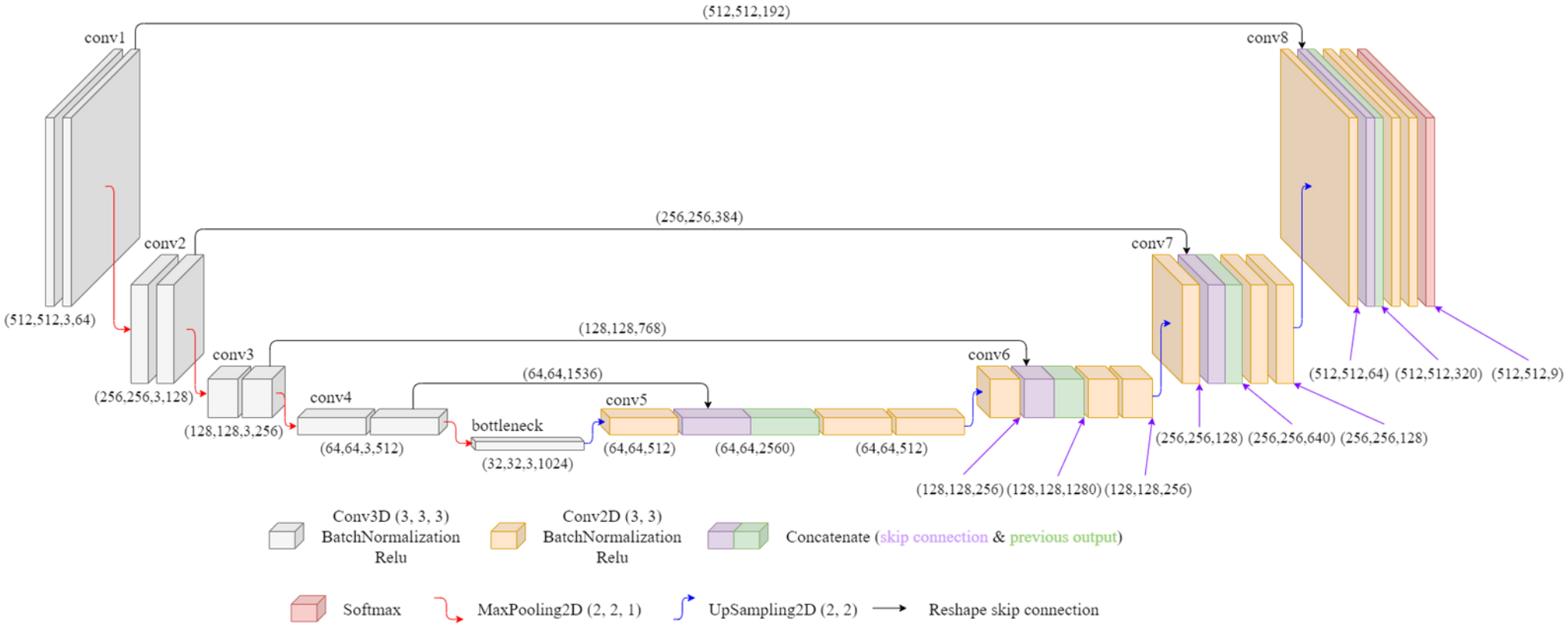

Figure S.I.4: Schematic of the architecture used for the 3D-2D U-Net model**.**

REFERENCES

[1] B. Miethig, A. Lui, S. Habibi, M.V. Mohrenschildt, “Leveraging Thermal Imaging for Autonomous Driving”, *IEEE Transportation Electrification Conference and Expo,* 2019.

[2] R. Sundar, S.C. Gupta, K.S. Verma, “Thermal Imaging: Reconnaissance & Surveillance Applications”, *Journal of Optics,* 1980.

[3] F. Amon, C.C. Pearson, “Experimental Methods in Physical Sciences” *Elsevier,* p.279-331, 2010.

[4] D. Perpetuini, C. Filippini, D. Cardone, A. Merla, “An Overview of Thermal Infrared Imaging-Based Screenings during Pandemic Emergencies”, *International Journal of Environmental Research and Public Health,* 2021.

[5] FLIR, “The Ultimate Infrared Handbook for R&D Professionals” *FLIR Systems, Inc,* 2012.

[6] N.K. Dhar, R. Dat, A.K. Sood, “Advances in Infrared Detector Array Technology”, *Optoelectronics,* 2013.

[7] M.F. Modest, “Radiative Heat Transfer,” *McGraw-Hill,* p.8-11, 1993.

[8] D. Zhao, A. Aili, Y. Zhai, S. Xu, G. Tan, X. Yin, R. Yang, “Radiative sky cooling: Fundamental principles, materials, and applications”, *Applied Physics Reviews,* 2019.

[9] L. Yu, Y. Guo, H. Zhu, M. Luo, P. Han, X. Ji, “Low-cost Microbolometer Type Infrared Detectors”, *Micromachines,* 2020.

[10] Fraunhofer IMS “Microbolometers as Uncooled Sensor Elements for Infrared Radiation” *Fraunhofer Institute for Microelectronics Circuits and Systems,* 2022.

[11] M. Abdel-Rahman, M. Hezam, A.A. Odebowale, N. Alkalli, M. Alduraibi, “TiNb thin films as absorbers for LWIR microbolometers”, *Optical Materials,* 2021.

[12] C. Bolakis, I. S. Karanasiou, D. Grbovic, C. Vazouras, G. Karunasiri, N. Uzunoglu, “Optimizing the Absorption Capability of a Microbolometer Pixel’s Active Element”, *International Journal of Electromagnetics and Applications,* 2019.

[13] H. Budzier, G. Gerlach, “Calibration of Infrared Cameras with Microbolometers” *Association for Sensors and Measurement,* 2015.

[14] Teledyne FLIR, “Use Low-Cost Materials to Increase Target Emissivity”, *Teledyne FLIR LLC,* 2015.

[15] Teledyne FLIR, “FLIR Research Studio Analysis Software”, *Teledyne FLIR LLC,* 2022.

[16] S. Indolia, A.K. Goswami, S.P. Mishra, P. Asopa, “Conceptual Understanding of Convolutional Neural Network- A Deep Learning Approach”, *Procedia Computer Science,* 2018.

[17] A. Esteva, K. Chou, S. Yeung, N. Naik, A. Madani, A. Mottaghi, Y. Liu, E. Topol, J. Dean, R. Socher, “Deep learning-enabled medical computer vision”, *NPJ Digital Medicine,* 2021.

[18] Z. Zhang, J. Geiger, J. Pohjalainen, A.E. Mousa, B. Schuller, “Deep Learning for Environmentally Robust Speech Recognition: An Overview of Recent Development”, *Computation and Language,* 2018.

[19] S.P. Singh, A. Kumar, H. Darbari, L. Singh, A. Rastogi, S. Jain, “Machine translation using deep learning: An overview”, *IEEE International Conference on Computer, Communication and Electronics,* 2017.

[20] H. Chen, O. Engkvist, Y. Wang, M. Olivecrona, T. Blaschke, “The rise of deep learning in drug discovery”, *Drug Discovery Today,* 2018.

[21] K. O’Shea, R. Nash, “An Introduction to Convolutional Neural Networks”, arXiv:1511.08458v2 [cs.NE] 2 Dec 2015.

[22] S. Vieira, W.H.L. Pinaya, A. Mechelli, “Using deep learning to investigate the neuroimaging correlates of psychiatric and neurological disorders: Methods and applications”, *Neuroscience & Biobehavioral Reviews,* 2017.

[23] N.J. Sairamya, L. Susmitha, S.T. George, M.S.P. Subathra, “Hybrid Approach for Classification of Electroencephalographic Signals Using Time–Frequency Images With Wavelets and Texture Features”, *Intelligent Data Analysis from Biomedical Applications,* 2019.

[24] V. Nair, G.E. Hinton, “Rectified Linear Units Improve Restricted Boltzmann Machines”, *Proceeding of the 27th International Conference on Machine Learning,* 2010.

[25] B. Shao, X. Hu, G. Bian, Y. Zhao, “A Multichannel LSTM-CNN Method for Fault Diagnosis of Chemical Process”, *Mathematical Problems in Engineering,* 2019.

[26] S. Albawi, O. Bayat, S. Al-Azawi, O.N. Ucan, “Social Touch Gesture Recognition Using Convolutional Neural Network”, *Computational Intelligence and Neuroscience,* 2018.

[27] R. Yamashita, M. Nishio, R.K.G. Do, K. Togashi, “Convolutional neural networks: an overview and application in radiology”, *Insights into Imaging,* 2018.

[28] H. Gholamalinezhad, H. Khosravi, “Pooling Methods in Deep Neural Networks, a Review”, *Computer Vision and Pattern Recognition,* 2020.

[29] R. Shanmugamani, “Deep Learning for Computer Vision”, *Packt Publishing,* 2018.

[30] J.S. Bridle, “Training Stochastic Model Recognition Algorithms as Networks can lead to Maximum Mutual Information Estimation of Parameters”, *British Crown,* 1990.

[31] F. Li, J. Johnson, S. Yeung, “Detection and Segmentation”, *Stanford University CS 231,* 2017.

[32] J.P. Vigueras-Guillen, B. Sari, S.F. Goes, H.G. Lemij, J.V. Rooij, K.A. Vermeer, L.J.V. Vliet, “Fully convolutional architecture vs sliding-window CNN for corneal endothelium cell segmentation”, *BMC Biomedical Engineering,* 2019.

[33] O. Ronneberger, P. Fischer, T. Brox, “U-Net: Convolutional Networks for Biomedical Image Segmentation,” *Medical Image Computing and Computer Assisted Interventions,* 2015.

[34] M. Pal, Akshay, H. Rohilla, B.C. Teja, “Patch Based Classification of Remote Sensing Data: A Comparison of 2D-CNN, SVM and NN Classifiers”, *Computer Vision and Pattern Recognition,* 2020.

[35] I. Tabian, H. Fu, Z.S. Khodaei, “A Convolutional Neural Network for Impact Detection and Characterization of Complex Composite Structures”, *Sensors (Basel, Switzerland),* 2019.

[36] H. Binol, “Ensemble learning based multiple kernel principal component analysis for dimensionality reduction and classification of hyperspectral imagery”, *Mathematical Problems in Engineering,* 2018.

[37] E.S. Mohamed, A.M. Saleh, A.B. Belal, A. Gad, “Application of near-infrared reflectance for quantitative assessment of soil properties”, *The Egyptian Journal of Remote Sensing and Space Science,* 2018.

[38] W. Lv, X. Wang, “Overview of Hyperspectral Image Classification”, *Journal of Sensors*, 2020.

[39] R. F. Kokaly, G. E. Graham, T. M. Hoefen, K. D. Kelly, M. R. Johnson, B. E. Hubbard, “Hyperspectral surveying for mineral resources in Alaska”, *U.S. Geological Survey*, 2016.

[40] M. Shimoni, R. Haelterman, C. Perneel, “Hypersectral Imaging for Military and Security Applications: Combining Myriad Processing and Sensing Techniques”, *IEEE Geosciences and Remote Sensing Magazine*, 2019.

[41] S. Trajanovski, C, Shan, P.J.C. Weijtmans, S.G. Brouwer de Koning, T.J.M Ruers, “Tongue tumor detection in hyperspectral images using deep learning semantic segmentation”, *IEEE Transactions on Bio-medical Engineering*, 2021.

[42] M. Schlerf, G. Rock, P. Lagueux, F. Ronellenfitsch, M. Gerhards, L. Hoffman, T. Udelhoven, “A Hyperspectral Thermal Infrared Imaging Instrument for Natural Resources Applications”, *Remote Sensing*, 2012.

[43] M.F. Baumgardner, L. L. Biehl, D. A. Landgrebe, “220 Band ARIVIS Hyperspectral Image Data Set: June 12, 1992 Indian Pine Test Site 3”, *Purdue University Research Repository*, 2015.

[44] A. Plaza, P. Martinez, J. Plaza, R. Perez, “Dimensionality reduction and classification of hyperspectral image data using sequences of extended morphological transformations”, *IEEE Trans. Geosci. Remote Sens.*, 2005.

[45] P. Gamba, “Pavia Center and University Data Sets”, *Pavia University Telecomm. and Remote Sensing Laboratory*, 2015.

[46] H. Zhang, L. Meng, X. Wei, Xi. Tang, Xu. Tang, X. Wang, B. Jin, W. Yao, “1D-Convolutional Capsule Network for Hyperspectral Image Classification”, *Computer Vision and Pattern Recognition,* 2019.

[47] T. H. Hseih, J.F. Kiang, “Comparison of CNN Algorithms on Hyperspectral Image Classification in Agricultural Lands”, *Sensing*, 2020.

[48] M. Ahmad, “A Fast 3D CNN for Hyperspectral Image Classification”, *Image and Video Processing*, 2020.

[49] S. K. Roy, G. Krishna, S. R. Dubey, B. B. Chaudhuri, “HybridSN: Exploring 3D-2D CNN Feature Hierarchy for Hyperspectral Image Classification*”, IEEE Geoscience and Remote Sensing Letters*, 2019.

[50] D. Lui, G. Han, P. Liu, H. Yang, X. Sun, Q. Li, “A Novel 2D-3D CNN with Spectral-Spatial Multi-Scale Feature Fusion for Hyperspectral Image Classification”, *Remote Sensing*, 2021.

[51] M.S.S. Moustafa, S.A. Mohamed, S. Ahmed, A.H. Nasr, “Hyperspectral change detection based on modification of UNet neural networks”, *Journal of Applied Remote Sensing*, 2021.

[52] C.A. Schneider, W.S. Rasband, K.W. Eliceiri, "NIH Image to ImageJ: 25 years of image analysis", *Nature Methods*, 2012.

[53] G. Bradski, “The OpenCV Library”, *Journal of Software Tools*, 2000.

[54] F. Zhuang, Z. Qi, K. Duan, D. Xi, Y. Zhu, H. Zhu, H. Xiong, Q. He, “A Comprehensive Survey on Transfer Learning”, *Machine Learning*, 2020.

[55] A.R. Zamir, A. Sax, W. Shen, L. Guibas, J. Malik, S. Savarse, “Taskonomy: Disentangling Task Transfer Learning”, *Conference on Computer Vision and Pattern Recognition,* 2018.

[56] E. Barreira, R. M. S. F. Almeida, M. L. Simoes, “Emissivity of Building Materials for Infrared Measurements”, *Sensors,* 2021.

[57] L. Goddijn-Murphy, B. Williamson, “On Thermal Infrared Remote Sensing of Plastic Pollution in Natural Waters”, *Environmental Remote Sensing,* 2019.

[58] B. Girardin, G. Fontaine, S. Duquesne, M. Forsth, S. Bourbigot, “Characterization of Thermo-Physical Properties of EVA/ATH: Application to Gasification Experiments and Pyrolysis Modeling”, *Materials,* 2015.