ABSTRACT

ZHU, HANCHI. A Neural Network Model to Predict the Color of Dry Cotton Fabric from a Wet State (Under the direction of Dr. Warren Jasper).

For most commercial dyes, water is used as the medium to enable dyes to disperse and diffuse evenly onto undyed fabrics during the dyeing process. However, the color of fabrics is generally viewed or measured when the fabric is dry. Although water itself is colorless, when a fabric is wet or viewed through water, its apparent color is different. For this reason, fabric samples must be dried before colorists can compare them with a target shade to determine if they match or not. The highly non-linear mapping in observed color in the dry and wet states has yielded poor predictive models. A three-layer neural network model was constructed, which predicted the color of dry cotton fabric from its wet state. Different models were developed based on squeeze pressure, with inputs to the models consisting of the L*a*b* reflectance values in the wet state and the outputs of the model consisting of the predicted L*a*b* values in the dry state. Without the neural net, the color difference ranged from 5 to 20 ΔE_{2000} color difference units. Using a neural net, the model correctly predicted the color of the dry fabric when measured in the wet state to under one color difference unit (using the ΔE_{2000} equation) over 90% of the time. The ΔE_{2000} color difference equation was used as a metric for training as well as for testing the model, as opposed to the more traditional Euclidian norm function. This resulted in faster model convergence and improved model accuracy,

A Neural Network Model to Predict the Color of Dry Cotton Fabric from a Wet State

by
Hanchi Zhu

A dissertation or thesis submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Fiber and Polymer Science

Raleigh, North Carolina
2022

APPROVED BY:

_____	_____
Dr. Warren Jasper Committee Chair	Dr. A. Godfrey
_____	_____
Dr. Arnab Maity	Dr. Lori Rothenberg

_____
Dr. Renzo Shamey

BIOGRAPHY

Hanchi Zhu is a Ph.D. student in Fiber & Polymer Science at North Carolina State University. His doctoral research involves using machine learning to create a model that can predict the color of dry cotton fabric while in a wet state. He has worked as a teaching assistant and lab technician under Jeffrey Krauss's supervision in the dyeing and finishing lab at the Wilson College of Textiles at North Carolina State University. He received his M.E. degree in Chemical Engineering from the University of Connecticut in 2017 and a B.S. degree in Textile Chemistry from Donghua University (China) in 2014. He also earned a M.S. degree in statistics at North Carolina State University in 2020.

ACKNOWLEDGMENTS

First and foremost, I would like to express my sincere appreciation to my major advisor Dr. Warren Jasper. I appreciate the opportunities and mentorship I received to pursue my Ph.D. degree in his group and am thankful for his support, help, patience, and guidance throughout my Ph.D. studies. Moreover, I really appreciate his help in teaching me how to conduct scientific research and his useful guidance in coding and color science. He also provided many suggestions for my career and personal life, which was an enormous help.

I also want to thank Dr. A. Blanton Godfrey, Dr. Arnab Maity, Dr. Peter Hasuer, Dr. Lori Rothenberg, and Dr. Renzo Shamey for being on my committee and thank them for their valuable time and suggestions in the fields of statistics, color science, and machine learning. Moreover, I am thankful to Dr. Andre West and Dr. Jon Rust for accepting me as a lab technician in the dyeing and finishing lab at the Wilson College of Textiles and funding me for my study. I want to especially thank Jeffrey Krauss for his care and lessons about all the lab and manufacturing experiences in dyeing and finishing. I cherish my time with all my lab mates in his lab.

Lastly, I want to thank the most important people in my life. Thanks to my parents, my younger sister, and my girlfriend for unconditioned love and support. They always encouraged me, whatever problems I confronted. Lastly, I would like to thank my best friends at NCSU for their friendship and the Wilson College of Textiles staff for their help and kindness. With them, I will never feel alone during my study abroad. Love you all forever.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Highlight and Contribution

This thesis describes an initial attempt to combine textile color science and neural network machine learning in order to create robust models to predict the color of dry fabrics from color measurements in the wet state. As such, these preliminary results should lead to a correct and formal methodology for future applications applicable not only to reactive dyes on cotton but to various types of fibers and dyes. In this project, I made the following contributions:

- Developed models to successfully predict the color of dry cotton fabric from measurements in their wet state. The models are accurate over 90% of the time in predicting the final color to under 1.0 color difference unit ΔE_{2000}
- Compared three color difference formulas as the error function (metrics) instead of the Euclidian norm function for Neural Network prediction models.
- Generated a methodology for designing and implementing AI algorithms to build a color prediction model in textile dyeing.
- Created a dyeing database consisting of 765 samples for future research and projects. Demonstrated how big data and data-intensive systems along with statistics and artificial intelligence are leveraged as powerful tools in textiles dyeing and finishing.

Chapter 1: Introduction

1.1. Color

The perception of the color results from the interaction between the light source, the surface of the object, and the stimulus-response of the eye-brain of an observer with normal color vision. (Marcus, 1997) The energy emitted from a light source is absorbed and reflected by the physical and chemical properties of the object's surface. The reflected light is detected by the photoreceptors of the eyes and generates an electrical signal to the brain. Our brain encodes the signal and produces a sensation of color. (Fairchild, 2013) Figure 1.1 below shows the interaction among these three basic components.

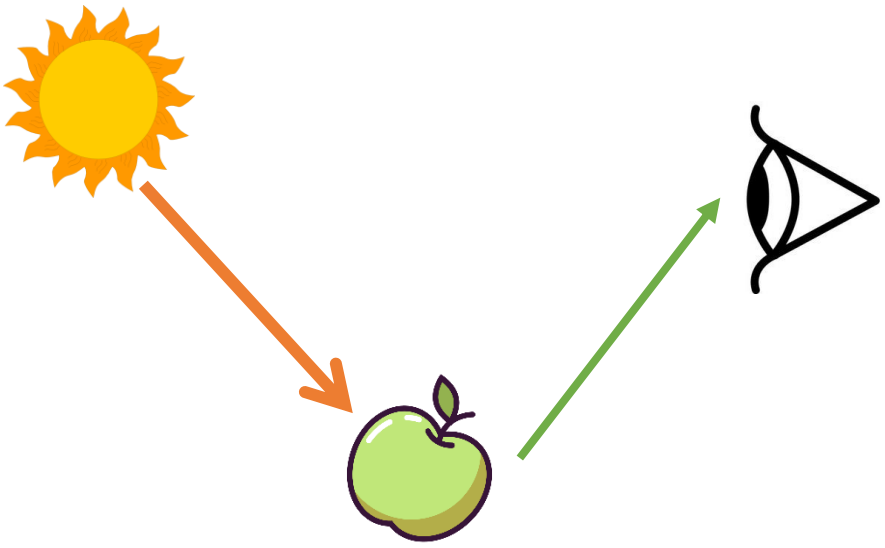


Figure 1.1. The three basic components for color perception

In order to measure and record the color, it is necessary to quantify these three components. Emitted light from a light source can be described by its spectral power distribution (SPD) across the visible spectrum. Objects can be specified according to their reflectance ratio in

comparison with a standard (for example a white tile) across the visible spectrum. The reflectance of a non-florescent object is constant at a given wavelength regardless of the light source, provided the object is illuminated by a continuum of emission. The reflectance of an object is modified by its surface geometry, surface texture, surface reflection (gloss), the amount of energy absorbed, scattered, or transmitted through the material, and other physical and chemical properties. Since the human eye and brain is individual and complex, the standard observation condition is an assumed average and will be discussed in the following section.

1.1.1. Illuminants and Light Sources

In 1672, Sir Isaac Newton demonstrated that white light or nearly white light can be decomposed into the visible spectrum using a prism. Visible light is a type of electromagnetic radiation detectable by the human eye and can be defined in terms of wavelength using a nanometer as the unit of length. (Berns, 2019; Cárdenas, 2009) Visible light is defined as electromagnetic energy in the range of 380 to 780 nm. Light from any light source can be quantified in radiometric terms and photometric terms. Radiometry may be defined as the science that studies the measurement of electromagnetic radiation, including visible light. Photometry may be defined as the science that studies the measurement of visible light weighted by the perception of brightness. (Marcus, 1997) Illuminance is related to the perception of brightness of a light source and is the amount of light falling on a surface weighted by a photometric brightness constant, and its unit is lux (lm m^{-2}).

The Commission Internationale de l'Eclairage (CIE) in 1931 quantifiably standardized three sources of illumination:

- Standard illuminant A was set to represent indoor artificial light with a correlated color temperature (CCT) of approximately 2856 K.
- Standard illuminant B represents direct sunlight and had a CCT of 4900 K.
- Standard illuminant C represents average daylight with a CCT of 6800K.

In 1963 a new set of recommendations were made by the CIE that defined the standard D illuminants, which represent various types of daylight under different conditions. (Marcus, 1997) In 2004, the CIE published a new report in colorimetry, which defines the basic recommendations concerning modern colorimetry. (CIE, 2004) Currently, the recommended illuminant standards are:

- Illuminant A: The CCT for illuminant A is 2856K, and it is used for colorimetric calculations in which incandescent illumination is needed.
- Illuminant B & C: These illuminants were intended to represent daylight at CCT of 4900 K and 6800 K. However, their use is no longer recommended by the CIE.
- Illuminant D series: The D series of illuminants are constructed to represent natural daylight. The most common illuminant to use for daylight is D65, which mimics noon daylight with a CCT of 6500K. Other illuminants are specified and used by different areas. For example, D55 mimics mid-morning daylight with a CCT of 5500K, and D75 mimics North Sky daylight with a CCT of 7500K.
- Illuminant E: Illuminant E is an equal-energy radiator and approximated with a CCT of 5455K.

- Illuminant series F: F series are fluorescent light sources. The most commonly used ones are F2 (cool white fluorescent lighting) and F11 (energy-efficient tri-band fluorescent lamps). (CIE, 2004)

For this project, we used D65 light source as the standard light source to test the color of cotton fabrics in the wet and dry states.

1.1.2. Standard Viewing Geometries

The perception of color is not only affected by the emitted light but is also defined by the viewing geometries. Viewing geometry is vital in cases in which the surface characteristics are determinant such as gloss properties. (Fairchild, 2013) Standard viewing geometries have been recommended by the CIE to be used in colorimetry trying to quantify the object's appearance effect based on reflectance in 2004, the most common of which are listed below. (CIE, 2004)

- Diffuse Eight-Degree Geometry Specular Component Included (di: 8º)
- Diffuse Eight-Degree Geometry Specular Component Excluded (de: 8º)
- 45º Annular/Normal Geometry (45º a:0º)
- Normal/ 45° Annular Geometry (0º:45º a)
- 45º Directional/Normal Geometry (45º:0º)
- Normal/ 45º Directional Geometry (0º: 45º)

1.1.3. Standard Observer

In the 1920's the CIE, while working to specify the red, green, and yellow colored lights used in railroads and later in road traffic lights, developed the concept of the standard observer and a standardized apparatus as a method for color specification. Guild and Wright, from separate research groups, measured the color matching functions by a set of 7 and 10 observers,

respectively. Both experiments were conducted under similar viewing conditions using filtered red, green, and blue lights, in which the stimulus intensity was recorded. The viewing field used in the experiments had a bipartite area limited to a 2° visual angle subtended on the retina, with the surround fixed with complete darkness. The experiment was based on Grassmann's laws of additive color matching. The observers were presented with a monochromatic spectral light on one side of the field, and they were required to match the color by combining and modifying the intensity of the three primary lights of red, blue, and green. (Berns, 2019)

In 1924 the CIE developed a system of photometry. The spectral luminous efficiency function, V(λ), was developed for photopic vision. Plotted as a function of wavelength, it shows that the human visual system is more sensitive to the perception of brightness in spectral wavelengths located in the middle of the visual spectrum and less sensitive towards the extremes. (Fairchild, 2013) It is understood that the V(λ) corresponds to a weighted sum of the three-cone response functions. Note that negative values exist in some ranges for the functions, as shown in Figure 1.2a. This presented problems to the CIE committee of the day, owing to the likely errors that would result from conducting many calculations by hand. To overcome this problem, the CIE transformed the RGB primaries to another set of primaries known as the XYZ coordinate system.

This imaginary set of primaries eliminated the negative values of the color matching functions. In addition, one of the color matching functions was forced to be equal to the photopic luminous efficiency function V(λ). The corresponding color-matching functions are known as $\overline{x_\lambda}$, $\overline{y_\lambda}$, and $\overline{z_\lambda}$. This system is known as the CIE 1931 2° standard observer (shown in Figure 1.2b). (Fairchild, 2013)

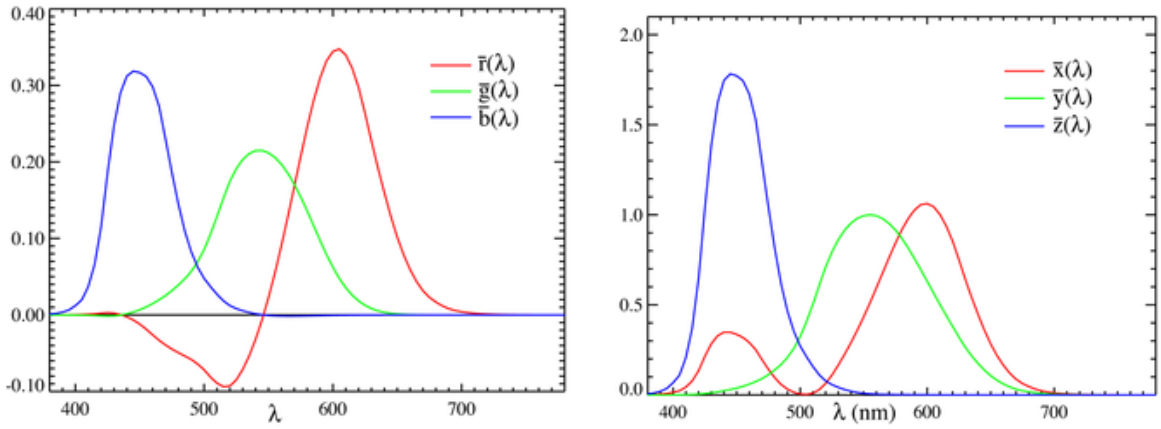


Figure 1.2. (a) Color matching functions for the 1931 standard observer (b) CIE 1931 2° standard observe. (Fairchild, 2013)

However, a large number of applications require larger visual fields of view, and as such, the CIE 1931 2° standard observer is not suitable. Therefore, the CIE determined that the 1931 2° standard observer would accurately predict matches only for a small visual field, and this may not be the case for large-field color matching owing to the dramatically differing distribution of cones as the angle subtended increases. Stiles, Burch, and Speranskaya attempted to figure out color matching functions for a 10 ° field of View. (Speranskaya, 1959; Stiles & Burch, 1959) Stiles and Burch designed the experiment with 49 normal observers and a high level of illumination to avoid rod intrusion, while Speranskaya used 27 observers and lower levels of illumination. The CIE removed the effect of rod intrusion in Speranskaya's data and combined the data from the two studies to develop what is known today as the 1964 CIE Supplementary Standard Observer, more commonly known as the 10º standard observer. The color matching functions are known as $\bar{x}_{10\lambda}$, $\bar{y}_{10\lambda}$ and $\bar{z}_{10\lambda}$. These color matching functions are mostly used when the visual field angle exceeds 4º, which is the case for many applications. (North & Fairchild, 1993a; North & Fairchild, 1993b; Randall, 1998) For our project, we used a 10º standard observer as the standard observer.

1.2. Color Space

With the help of color measurement instruments such as reflectance spectrophotometers, the apparent color of an object can be quantified from its reflectance by three real numbers in a 3-D coordinate or vector space. A color space is a mathematical vector space describing the way colors can be represented as points in three-space. A given color model is defined by the specific mapping function according to the reference color space. The shape of a reference color space in color space coordinates is known as a gamut. Commonly, standard color spaces are the CIELAB color space or CIEXYZ color space, which were designed to encompass almost all colors that can normally be observed. (Sharma & Rodrguez-Pardo, Jan 23, 2012)

In 1802, before people discovered cone cells in the human eye, Thomas Young postulated there were three types of receptors in the retina, each of which was sensitive to a particular range of visible light and suggested that the human color vision should be trichromatic. In 1850, Hermann Helmholtz found that three types of cone photoreceptors could be classified as short-wavelength (blue), middle-wavelength (green), and long-wavelength (red) due to their response. Those three cone cells would generate and transport signals into the brain after receiving a particular wavelength of light. (Janos, 1996) Color blindness was explained by the absence of one or more types of cone cells.

About the same time as trichromatic theory developed, Ewald Hering, hypothesized a different mechanism of color vision; Hering's opponent color theory postulated that there are three sets of receptors, one of which is sensitive to white and black, another to red and green, and the third to yellow and blue. (*Opponent-process theory of colour vision.* 2009; Hurvich & Jameson, 1957) The breaking down of these substances is supposed to yield one member of these pairs (white, red, or yellow), while the building up of the same substances yields the other (black,

green, or blue). Color blindness was explained by the absence of one or more substances in the chromatic processes. (Cárdenas, 2009; Coren et al., 2003; Mark D. Fairchild, 2013)

In 1853, Hermann Grassmann pointed out the vector character of color. Color, although a psychological phenomenon, can be modeled mathematically using algebraic and geometric concepts. Grassmann also published a theory of how colors mix, which is known as Grassmann's law. (Fearnley-Sander, 1979)

The CIE 1931 XYZ color space was one of the first attempts to map the color space based on the measurement of human perception. The color-matching properties of the CIE 1931 standard observer, defined by the color-matching functions: $\bar{x}(\lambda), \bar{y}(\lambda)\ and\ \bar{z}(\lambda)$, were defined in the wavelength range from 380nm to 780nm with an interval of 5nm. Later the CIELUV color space was developed from the CIE 1931 XYZ color space. Because of its more visually uniform distribution of colors, the color differences could be more easily determined. In 1964, as a supplement to the CIE1931 color space, the CIE recommended an alternative set of standard color-matching functions: $\bar{x}_{10}(\lambda), \bar{y}_{10}(\lambda)\ and\ \bar{z}_{10}(\lambda)$. These new functions defined a new ideal observer with 10 degrees as the viewing angle. Compared to a 2-degree field, color matching could be estimated to be two or three times as precise. In 1976, CIELAB produced a color space that was even more perceptually uniform than other color spaces. Perceptually uniform means that a change of the same amount in a color value in any direction should produce a change of about the same visual importance. Our research is based on the CIELAB color space. (Wyszecki & Stiles, 1982)

There are also many other color spaces whose uses depend on particular applications. HSV color space is good at accurately detecting the edges of images and determining the position for the prediction of image movement. (Guan, 2018; Radovan & Ban, May 2018) The

CMYK color model, a subtractive model based on the CMY color model, is used in the printing industry because it quantifies the reflected light of inks. The Munsell color system is a color space based on three properties of color: hue, chroma, and value. The Natural Colour System (NCS) is based on three pairs of opponent colors, defined by the Hering theory of color vision. For better communicating and reproducing the same color, the Pantone matching system was created by Pantone LLC. This color space is used widely from design labs all the way to production mills. (Bello-Cerezo et al., 2016)

1.3. Color difference

One of the most important applications of color spaces is for the calculation of color differences. If one assumes that CIELAB to be a perfectly perceptually uniform color space (which it is not), then the magnitude of the distance between any two colors (points) will correlate with the magnitude of the color difference visually perceived. Therefore mathematically, the differences in lightness (ΔL^*), redness-greenness (Δa^*), yellowness-blueness (Δb^*), chroma (ΔC^*), and hue (ΔH^*) could be calculated and compared. An accurate mathematical model of a uniform color space must either closely represent perceived colors or have sufficient variables to model the relationship between perceptual and physical data accurately. Presently, there is no single model or metric which represents the mechanism of perceived color exactly.

In a completely uniform color space, the tolerance boundary (Pass/Fail tolerance boundaries) can be represented geometrically as a sphere. Therefore, the same perceived difference in lightness would correspond to the same distance as an equal difference in perceived chroma or hue. In 1976, CIE 76 was the first formula that combined color difference and a

known color space. (Heggie et al., 1996; Li, P., Wang, & Jing, 2015a; Wardman et al., 2012) From the CIELAB color space, two colors or points in three-space can be compared for visual color difference. The CIE76 formula is given as:

$$\Delta E^*_{ab} = \sqrt{\left(L^*_2 - L^*_1\right)^2 + \left(a^*_2 - a^*_1\right)^2 + \left(b^*_2 - b^*_1\right)^2} \quad (1.1)$$

Since the CIELAB color space is not perceptually uniform, the true tolerance boundary will not be spherical in practice. The CIELAB ΔE_{76} formula predicted more incorrect decisions of pass/fail than the average expert visual assessor. Some companies and researchers worked to develop modified color difference formulas that would be more accurate than ΔE_{76}. (Mangine et al., 2005; Ugur & Behcet, 2019)

The CIE1994 formula was developed to address perceptual non-uniformities while retaining the CIELAB color space by the introduction of application-specific weights. (McDonald & Smith, 2008) k_C and k_H in the color difference formula are usually both unity weighting factors, while k_L, K_1 and K_2 depend on the application. For textiles, they are 2, 0.048 and 0.014 respectively.

$$\Delta E^*_{94} = \sqrt{\left(\frac{\Delta L^*}{k_L S_L}\right)^2 + \left(\frac{\Delta C^*_{ab}}{k_C S_C}\right)^2 + \left(\frac{\Delta H^*_{ab}}{k_H S_H}\right)^2} \quad (1.2)$$

Where,

$$
\begin{aligned}
&\Delta L^* = L_1^* - L_2^* \\
&C_1^* = \sqrt{a_1^{*2} + \ b_1^{*2}} \\
&C_2^* = \sqrt{a_2^{*2} + \ b_2^{*2}} \\
&\Delta C_{ab}^* = C_1^* - C_2^* \\
&\Delta H_{ab}^* = \sqrt{\Delta E_{ab}^{*\ 2} - \ \Delta L^{*2} - \Delta C_{ab}^{*\ 2}} = \sqrt{\Delta a^{*2} - \ \Delta b^{*2} - \Delta C_{ab}^{*\ 2}} \\
&\Delta a^* = a_1^* - \ a_2^* \\
&\Delta b^* = b_1^* - \ b_2^* \\
&S_L = 1 \\
&S_C = K_1 C_1^* + 1 \\
&S_H = K_2 C_1^* + 1
\end{aligned}
\tag{1.3}
$$

Since the CIE ΔE^*_{94} formula still did not solve the issue of uniformity, the CIE committee modified their color different formula by adding five additional corrections. (Heggie et al., 1996; Li, P., Wang, & Jing, 2015b) The CIE ΔE_{2000} formula attempts to correct for errors in perceived color difference that depend on the absolute color position and not just on color differences. The reference color is therefore very important. In ΔE_{2000}, the a* value is determined uniquely for each pair of colors. Thus, ΔE_{2000} is optimized for pairwise comparison but not fit for statistical control. In our project, predicted color and actual measurement color would be compared by a one-to-one method. Thus, ΔE_{2000} is one of the metrics we would consider to judge our models. (Luo, M. R. et al., 2001; Melgosa et al., 2017; Oulton & Westland, 2017)

$$
\Delta E_{2000}^* = \sqrt{\left(\frac{\Delta L'}{k_L S_L}\right)^2 + \left(\frac{\Delta C'}{k_C S_C}\right)^2 + \left(\frac{\Delta H'}{k_H S_H}\right)^2 + R_T \frac{\Delta C'}{k_C S_C} \frac{\Delta H'}{k_H S_H}}
\tag{1.4}
$$

Where,

$$\Delta L' = L_2^* - L_1^*$$

$$\bar{L} = \frac{L_2^* + L_1^*}{2}$$

$$\bar{C} = \frac{C_2^* + C_1^*}{2}$$

$$a_1' = a_1^* + \frac{a_1^*}{2}\left(1 - \sqrt{\frac{\bar{C}^7}{\bar{C}^7 + 25^7}}\right)$$

$$a_2' = a_2^* + \frac{a_2^*}{2}\left(1 - \sqrt{\frac{\bar{C}^7}{\bar{C}^7 + 25^7}}\right)$$

$$C_1' = \sqrt{{a_1'}^2 + {b_1^*}^2}$$

$$C_2' = \sqrt{{a_2'}^2 + {b_2^*}^2}$$

$$\Delta C' = C_2' - C_1'$$

$$\bar{C}' = \frac{C_2' + C_1'}{2}$$

$$h_1' = \text{atan2}(b_1^*, a_1') \quad mod\ 360^\circ$$

$$h_2' = \text{atan2}(b_2^*, a_2') \quad mod\ 360^\circ$$

$$\Delta h' = \begin{cases} h_2' - h_1' & |h_2' - h_1'| \le 180^\circ \\ h_2' - h_1' + 360^\circ & |h_2' - h_1'| > 180^\circ, h_2' \le h_1' \\ h_2' - h_1' - 360^\circ & |h_2' - h_1'| \le 180^\circ, h_2' > h_1' \end{cases}$$

$$\bar{H}' = \begin{cases} (h_2' + h_1')/2 & |h_2' - h_1'| \le 180^\circ \\ \dfrac{h_2' + h_1' + 360^\circ}{2} & |h_2' - h_1'| > 180^\circ, h_2' + h_1' < 360^\circ \\ (h_2' + h_1' - 360^\circ)/2 & |h_2' - h_1'| > 180^\circ, h_2' + h_1' \ge 360^\circ \end{cases}$$

$$\Delta \bar{H}' = 2\sqrt{C_1' C_2'}\sin\left(\frac{\Delta h'}{2}\right)$$

$$T = 1 - 0.17\cos(\bar{H}' - 30^\circ) + 024\cos(2\bar{H}') + 0.32\cos(3\bar{H}' + 6^\circ) - 0.20\cos(4\bar{H}' - 63^\circ)$$

$$S_L = 1 + \frac{0.015(\bar{L} - 50)^2}{\sqrt{20 + (\bar{L} - 50)^2}}$$

$$S_C = 1 + 0.045\bar{C}'$$

$$S_H = 1 + 0.015\bar{C}'T$$

$$R_T = -2\sqrt{\frac{\bar{C}^7}{\bar{C}^7 + 25^7}}\, sin\left[60^\circ * \exp\left(-\left[\frac{\bar{H}' - 275^\circ}{25^\circ}\right]^2\right)\right]$$

Another metric studied in our project is CMC (l:c). In 1984, the color measurement committee of the society of dyes and colorists defined another color difference measurement method CMC (l:c), based on the L*a*b* color space. (CMC colour-difference formula.1984; AATCC, 2015; Kulappurath, 2018; Luo, M. R. & Rigg, 1986; Shamey & Zhao, 2014) Two invoked parameters, lightness (l) and chroma (c), allow for a weighted difference based on the ratio of l:c, which depends on the application. Commonly used values are 2:1 for acceptability and 1:1 for the threshold of imperceptibility. The CMC tolerance ellipsoid represents the volume of acceptance in relation to the standard. The size of the CMC tolerance ellipsoid varies depending on its position in the color space. In the textile industry, people set the l:c value to 2:1, because the eye is more sensitive to chroma than lightness. Thus, we will use CMC as one of the color difference criteria.

$$\Delta E^*_{CMC} = \sqrt{\left(\frac{L^*_2 - L^*_1}{lS_L}\right)^2 + \left(\frac{C^*_2 - C^*_1}{cS_C}\right)^2 + \left(\frac{\Delta H^*_{ab}}{S_H}\right)^2} \qquad (1.5)$$

Where,

$$
\begin{aligned}
S_L &= \begin{cases} 0.511 & L_1^* < 16 \\ \dfrac{0.040975 L_1^*}{1 + 0.01765 L_1^*} & L_1^* \geq 16 \end{cases} \\
S_C &= \frac{0.0638 C_1^*}{1 + 0.0131 C_1^*} + 0.638 \\
S_H &= S_C \left(FT + 1 - F \right) \\
F &= \sqrt{\frac{C_1^{*4}}{C_1^{*4} + 1900}} \\
T &= \begin{cases} 0.56 + \left| 0.2 * \cos(h_1 + 168^\circ) \right| & 164^\circ \leq h_1 \leq 345^\circ \\ 0.36 + \left| 0.4 * \cos(h_1 + 35^\circ) \right| & \text{otherwise} \end{cases} \\
\Delta H_{ab}^* &= \sqrt{\Delta a^{*2} + \Delta b^{*2} + \Delta C_{ab}^{*\,2}} \\
\Delta C_{ab}^* &= C_1^* - C_2^* \\
a_1^* &= C_1^* \cos(h_1^*) \\
a_2^* &= C_2^* \cos(h_2^*) \\
b_1^* &= C_1^* \sin(h_1^*) \\
b_2^* &= C_2^* \sin(h_2^*) \\
\Delta a^* &= a_1^* - a_2^* \\
\Delta b^* &= b_1^* - b_2^*
\end{aligned}
\tag{1.6}
$$

It is difficult for most people to quantify the color difference by visual inspection, and therefore a range of color difference values are associated with a perceived observation. In Table 1.1, Brainard gives the meaning of some levels of color difference. (Brainard, 2003) For a better understanding, there are five pairs of colors (in Figure 1.1.) representing each of the five levels shown in Table 1.1.

Table 1.1. Level of color difference

Delta E	Perception
<= 1.0	Not perceptible by human eyes.
1 - 2	Perceptible through close observation.
2 - 10	Perceptible at a glance.
11 - 49	Colors are more similar than opposite
100	Colors are exact opposite

1.4. Kubelka-Munk Theory & Goldfinger's Model

The perceived color of a fabric is determined by observed reflected light emanating from its surface. In 1931, Kubelka and Munk introduced a mathematical model to describe the relationship between absorbed and reflected photons in the electromagnetic spectrum. Kubelka-Munk theory is a simple but successful theory to describe the reflectance and absorbance of light. It mostly explains the relationship between the absorption and scattering properties of fabrics. (Kubelka & Munk, 1931; Paul Kubelka & Franz Munk, 1931) The basic expression of the equation is given below as:

$$\frac{K_\lambda}{S_\lambda} = \frac{(1-R_\lambda)^2}{2*R_\lambda} \tag{1.6}$$

Where, K_λ is the absorption coefficient,

S_λ is the scattering coefficient,

R_λ is the reflectance of the material

λ is the wavelength of the light.

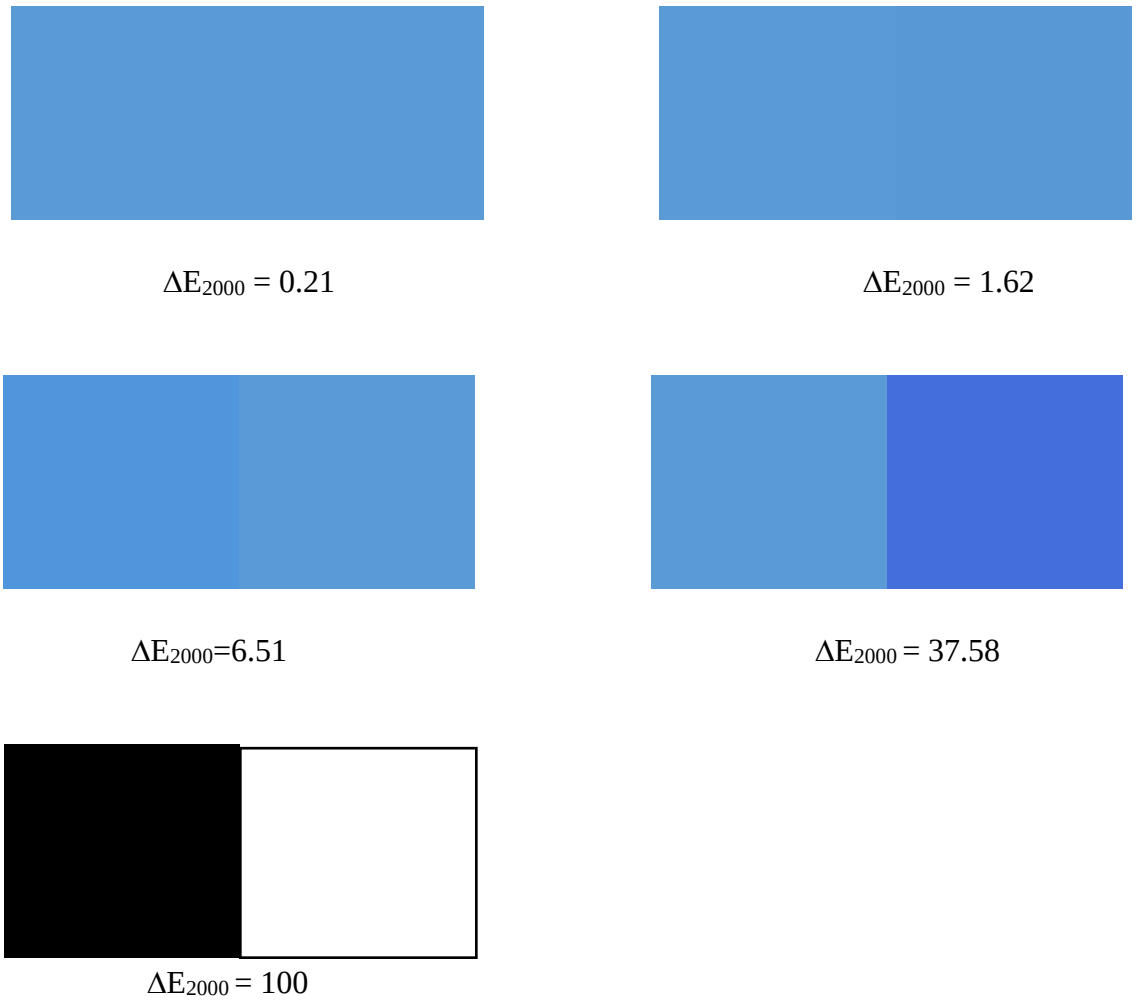


Figure 1.3. Color Difference Example

(*PANTONE textile color guide*; *PANTONE textile color specifier*)

This theory deals with monochromatic light, but due to the additive nature of light waves generalizes to polychromatic light. The tested colored materials are assumed to be sufficiently large so that a measurable amount of diffuse light will be reflected from the material's surface. The surface of the tested materials is assumed to be uniform, homogenous, and infinitely thick so that all the incident light is assumed to be absorbed or scattered (none is transmitted through the

fabric). The reason Kubelka-Munk theory is widely used is that it is fairly easy to calculate the ratio of K_λ / S_λ given the surface reflectance R_λ. In practice, K_λ and S_λ are two wavelength-dependent functions that are very difficult to calculate, but whose ratio is easy to measure. However, there are limitations to the theory such as:

- The illumination and collection fluxes in current color measurement instrumentation are not diffuse.
- The texture and the optical properties of the materials are not accounted for in the model.
- The remission factor K_λ / S_λ ratio is actually optimized to improve the behavior of the theory, so the resulting K_λ and S_λ coefficients do not exactly correspond to the physical coefficients of absorption and scattering.
- The size of the tested materials may be smaller than the incident light wavelength only to satisfy the assumption that half of the flux is scattered forward and half backward.
- The materials could have different textures when viewed from different angles. For example, fabrics have different linear densities due to variations in yarn diameters or construction. With different diameters, the color of the fabric looks different because of different reflectance values for the same dye concentration.
- The reflectance of the material changes as well if there are some embedding medium surrounding components on the surface. For instance, color changes may occur when dry fabrics become wet.
- Kubelka-Munk theory fails to explain extreme cases when the reflectance is very high or very low.

- Fluorescence is not accounted for in Kubelka-Munk theory. This theory assumes that there is no interaction between different wavelengths (linear independence).

Based on the limitations above, the Kubelka-Munk theory was improved by introducing new coefficients and developing new models such as a ray-tracing model. Duntley introduced the concept of a separate absorption coefficient for the directional irradiation to the absorption coefficient for the initial diffuse light according to the original Kubelka-Munk theory. Duntley's theory avoids the logical objections that apply to the two-constant theory developed by Kubelka and Munk, and other contributors. There are 5 coefficients to account for directional illumination with a different scattering in the forward and backward directions. (Duntley, 1942; Lathrop, 1965) Pineo developed a gloss equation similar to the Kubelka-Munk remittance expression, known as the Pineo correction equation. This equation is still in use to correct the gloss on textiles by incorporating empirical coefficients. (Li, S. et al., 2009; Pineo, 1940) Another significant work was made by Atherton. He brought the idea of light paths inside the fabrics and light path distribution patterns to the Kubelka-Munk theory. (Atherton, 1955; Garrett & Peters, 1956) Other developments were created and expanded based on the Kubelka-Munk theory. The most extensive multi-flux model was developed by Mudgett and Richards. However, their model is too complicated to handle and calibrate because of a large number of initial parameters. (Brinkworth, 1972; Mudgett & Richards, 1971; Mudgett & Richards, 1972)

A ray tracing model was created in an attempt to describe the light path in the structure of the material, according to the principles of optics. Fothergill developed an empirical equation related to textiles. He published an equation to deal with the distributions of the linear densities of fibers. He pointed out that polymer and dye orientation, as well as preferential light

polarization absorption by certain dye categories, would affect the fiber shade. (Athanasios A. Tsoutseos et al., 1999; Petrulis, 2014) Garrett and Peters combined the Kubelka-Munk theory and their own work. With a ray tracing model, light paths were observed on cylindrical fibers. Their goal was to model the change in fiber color due to different dye penetration levels, which is the effect of ring dyeing on textiles. (Garrett & Peters, 1956) Another important step in a ray tracing model was made by Melamed, who suggested that colored powder could be considered as spheres and the light paths were calculated using the basic principles of geometric optics. (Garay et al., 2004; Melamed, 1963) Some light rays were reflected by the powder, while other light rays passed directly through the powder and were reflected by the bottom substrate below the powder. The powder diameters were larger than the light wavelength, introducing errors to the original Kubelka-Munk Theory. (Athanasios A. Tsoutseos & James H. Nobbs, 1999; Athanasios A. Tsoutseos et al., 1999; Kubelka & Munk, 1931; Paul Kubelka & Franz Munk, 1931)

In the 1970s, Allen and Goldfinger developed the first ray-tracing model for textile fibers. (Allen, E. H. & Goldfinger, 1971; Allen, E. Hope & Goldfinger, 1972) This model was inspired by the Stock model using the idea of a 'pile of plates'. (Athanasios A. Tsoutseos & James H. Nobbs, 1999) In their model, the textile material was considered as individual cylindrical fibers, which were isotropic in structure and color and had the same diameter. Their diameter was assumed large compared to the wavelength of light, parallel to each other and stacked as an array. Each array is a 'plate' in the Stock model and stays in the same medium, mostly the air. The prediction of the light paths could be done by applying optics, physical background, fiber diameter, refractive index, the dye extinction coefficient, and the refractive index of the inside medium. The light path in a fiber cross-section is shown below:

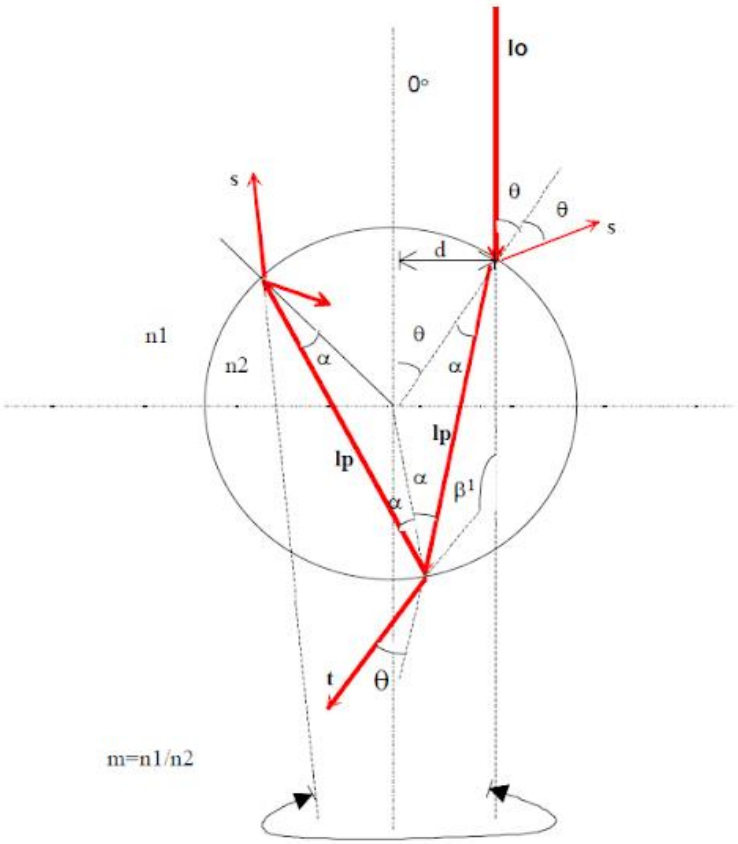


Figure 1.4. Light propagation in a fiber cross-section (Allen & Goldfinger, 1972)

Where, I_0 is the incident radiation

θ is the angle of light incidence on the fiber surface

d is the distance of the point of incidence from the vertical to the 0°

s is any light that travels in the upward direction

t is any light the travels in the downward direction

lp is the light path inside the fiber, before refraction

n_1 is the refractive index of the fiber

n_2 is the refractive index of the medium

m is the ratio of the refractive indices

α, β are the internal angles characterizing the refraction/reflection direction.

As shown in Fig. 1.2, when the initial light reaches the surface of the fiber, part of it is reflected, and the rest of it diffracted inside the fiber. After reaching the other inside surface of the fiber, part of the light was diffracted to the outside of the fiber, and the rest of it continued propagating inside the fiber until it was completely diffracted out. The most important fact of this model is that it is successful in combining geometric and optical properties of the light, the surrounding medium, the extinction coefficient of the colorant, and the optical properties of the fiber together to mimic the reflectance of the textile fiber.

1.5. Machine Learning

What is Machine Learning? In simple terms, machine learning is a method of training an artificial neural network to characterize inputs to outputs without explicit programming. Computers generally behave like finite state machines with predictable responses to known inputs. A more formal definition of machine learning is the application of artificial intelligence that provides systems the ability to automatically learn and improve from experience without being explicitly programmed. Machine learning focuses on the development of computer programing that access more and more data and "learn" for itself. Machine learning lets computers mimic and think like a human brain.

Learning rules are algorithms for finding suitable weights and/or other parameters. Currently, there are four major machine learning methods. (Jebara, 2003) They are supervised machine learning algorithms, unsupervised machine learning algorithms, semi-supervised machine learning algorithms, and reinforcement machine learning algorithms. Supervised machine learning algorithms predict target variables from data learned in the past. Beginning

from the analysis of an existing and an actual training dataset, the system creates an inferred algorithm to predict the target output values. The inferred algorithm can be corrected and validated by incorporating new data with the existing dataset. With the help of sufficiently large amounts of data, the inferred algorithm would improve and predict more accurate target values. Thus machine learning is good at non-linear interpolation but has difficulty in extrapolation. Alternatively, an unsupervised machine learning algorithm lets the system learn without any special output. The system does not classify or train the input data, but rather it discovers patterns from the data itself and draws inferences to figure out hidden structures and relationships between the data. For example, people who always smile can be grouped as happy, and those who frown can be grouped as sad. Semi-supervised machine learning algorithms combine both supervised and unsupervised machine learning algorithms by attempting to both improve the accuracy of the predicted values and draw conclusions and relationships from the input data. Usually, in semi-supervised machine learning, the most important dataset is selected first, and later less important datasets are selected. Reinforcement machine learning algorithms mostly apply in the area of video games. It allows the machine and software agents to maximize their performance by the ideal behavior through the environment and specific context. Trial and error search and delayed reward are the most relevant characteristics of reinforcement machine learning. Simple reward feedback is required for the machine to learn which action is best; this is known as the reinforcement signal.

These four machine learning methods described above are classified by their learning style. If machine learning algorithms are instead grouped by their work functions, they can be classified as follows:

1. Regression algorithms minimize the errors of each prediction. These algorithms are based on statistics. An example would be performing a linear regression on a set of data points. The sum of the distance between the data points and the regression model line constitutes an error that would be minimized. The most popular regression algorithms are linear regression, ordinary least squares regression, logistic regression, stepwise regression, and multivariate adaptive regression splines. (Giuseppe, 2017)
2. Instance algorithms attempt to cluster or arrange data that are deemed important or required for the model. It builds its own database and updates the database to create the best match compared with the old database. Unless data is pruned, the size of the database can grow exponentially and is a potential weakness of these algorithms. Some common instance-based algorithms are k-Nearest Neighbor (known as kNN), Learning Vector Quantization, self-organizing map, and support vector machines. (Cui & Gong, 2018; Shataee et al., 2012)
3. Decision tree algorithms create a decision tree net due to the actual values of attributes in the data. This model is usually constructed and trained by classification and regression of the raw data. Decision tree models are fast, accurate, and a favorite in machine learning. Some popular decision tree models are classification and regression tree (CART), Iterative Dichotomiser 3 (ID3), Chi-squared automatic interaction detection (CAID), and conditional decision tree. (Baghdadi et al., 2021; Quinlan, 1986)
4. Clustering algorithm is a classification algorithm. This method is typically organized by the modeling appearances such as centroid-based and hierarchical. All the methods are considered to use the inherent structures in raw data to best organize and separate

the data into groups of maximum commonality. Major clustering algorithms are k-means clustering, k-median clustering, Expectation Maximizing (EM), and hierarchical clustering. (Kim et al., 2021; Zheng et al., 2021)

The next common algorithms are dimensional reduction algorithms. In an arbitrary dataset with one dependent variable and many independent variables, some independent variables are robust and important to the data analysis, and some are not related to the final prediction. In order to speed up the processing time and work more efficiently, dimension reduction becomes a powerful and helpful method. Dimensional reduction algorithms seek to exploit the inherent structure in the raw data, but compared to clustering, it summarizes and describes the data using less information in an unsupervised manner. This can be useful to visualize dimensional data or to simplify data. Many of these methods can be adapted for use in classification and regression. The most common reduction methods are listed below.

- Principal Component Analysis (PCA)
- Principal Component Regression (PCR)
- Partial Least Squares Regression (PLSR)
- Sammon Mapping
- Multidimensional Scaling (MDS)
- Projection Pursuit
- Linear Discriminant Analysis (LDA)
- Mixture Discriminant Analysis (MDA)
- Quadratic Discriminant Analysis (QDA)
- Flexible Discriminant Analysis (FDA)

Artificial neural networks (ANN) and deep learning algorithms (DLA) are the techniques used in this research. Artificial neural networks, as the name suggests, were inspired to mimic biological neural networks. Deep learning algorithms actually are an upgraded version of artificial neural networks, defined as neural networks architectures that can facilitate deep learning, retrieval, and analysis of data that is deeply buried in input information and not otherwise easily retrievable. Their ability to dig deeply into the input data is often superior and/or faster than other computational methods due to their efficient integration of several and often many mathematical logical and computational methods; linear or nonlinear, analytic or heuristic, and deterministic or stochastic. Artificial neural networks are a class of pattern matching algorithms that are commonly used for regression and classification problems and are also employed in many subfields comprised of hundreds of algorithms and variations for all manner of problem types. Deep learning algorithms are composed of many complex neural networks and very large datasets. Deep learning methods usually are applied in the area of image, audio, and video processing. Some popular artificial neural networks are perceptron, back-propagation, multilayer perceptron, Hopfield network, Stochastic Gradient Descent, and Radial Basis Function Network. (Jawahar et al., 2015; Kan et al., 2016; Khataee & Mirzajani, 2010; Shen et al., 2017) The major deep learning algorithms are convolutional neural networks (CNN), recurrent neural networks (RNNs), Long Short-term memory networks (LSTMs), Deep Boltzmann Machine (DBM), and Deep belief networks (DBN). In relation to deepness and performance of learning, these can be classified into Back Propagation Neural Network, CNN, Large Memory Storage and Retrieval Neural Network, DBM, RNNs, and Deconvolution/Wavelet Neural Network (DCNN/WNN). (Li, H. et al., 2015) It should be pointed out that BP has a very broad range of applications but relatively slow convergence,

although CNN is the most popular algorithm, especially in the modeling of RNN and DCNN (or WNN). (Veit, 2021) Neural Network is a strong and popular tool to create prediction models in many areas. It can be used for 13C NMR chemical shifts prediction in chemistry and physical properties prediction of the steel alloys after plasma nitriding. (Kiryanov et al., 2018; Pribbenow et al., 2019) It can also help to process weather and climate forecast and predict the power outage for the utilities and insurance companies combined with the geographical acknowledges. (Yang, Wanik et al., 2020; Yang, Watson et al., 2020; Yang et al., 2021) In textiles, neural networks have been applied to detect defects in fabrics. (Jun et al., 2021; Li, C. et al., 2020; Zhao, S. et al., 2020; Zhao, X. et al., 2021) Also, with the help of statistical analysis, they can be used to forecast the demand market of textiles. (Lorente-Leyva et al., 2021)

1.6. Artificial Neural Networks & Color prediction in textile

Artificial Neural Networks (ANN) models can be applied to many applications in color prediction in textiles. (Furferi et al., 2016; Jawahar et al., 2015; Khataee & Mirzajani, 2010; Shen et al., 2017) Currently, colorists achieve a target color by attempting different dye formulas, which will eventually converge to the desired shade. Many color models are based on heuristics, or in other words, some color situations cannot be explained completely by our current knowledge, such as the color in fabrics that would be changed in different mediums. A neural network is a promising tool that can help in this area because of the non-parametric nature of neural networks. A neural network model can predict the target color by statistics without knowing the fundamental physical theories. (Saeed et al., 2014; Shams-Nateri et al., 2006) In one of the first applications of neural networks to coloration, Jasper used a neural network to predict the dye concentration in multiply-dye mixtures from their spectrophotometric absorbance

based on Beer's Law. The model created was successful in reducing the relative errors from 69% down to 9%. (Jasper, Warren J. et al., 1993) In the following year, Jasper also used a neural network combined with NIR spectrophotometry to identify different types of fibers by their IR absorbance spectra, which was another contribution to the textile field. (Jasper, W. J. & Kovacs, 1994) Currently, CNN models are used to identify cashmere and wool by imaging. (Luo, J. et al., 2021)

Neural networks are widely used in textiles, and an increasing number of research projects are applying neural networks as their support tools for better results. (Artificial Neural Network In Fabric Engineering.2012) In 1999, Athanasios's group tried to improve the accuracy of color matching with the combination of a novel color appearance model and a neural network and described the benefits of using a neural network prediction model to the dye recipe match compared to the common color match recipe. (Athanasios A. Tsoutseos & James H. Nobbs, 1999) The neural network model not only works for cotton fabrics but also can be applied to nylon, lycra, polyester, and various blends & pre-treated fabrics. (Balci et al., 2008; Hemingray & Westland, 2016; Hung, O. et al., 2017; Hung, O. N. et al., 2011; Hung, O. N. et al., 2017; Kan et al., 2016; Kuo & Fang, 2006; Xiao et al., 2020) Recently, ANN has been combined with regression or classification models with applications to textile printing, such as inkjet printing and digital printing for better color precision. (Hajipour & Shams-Nateri, 2019; Hwang et al., 2015) Almodarresi et al. used a scanner instead of a spectrophotometer to do color matching experiments with the help of neural network techniques. (Almodarresi et al., 2013) Furferi et al. predicted the color and color solidity of a jigger-dyed cellulose-based fabric by cascade neural network method and the results showed a small CIELAB distance and a predictable color solidity for the test fabric. (Furferi & Carfagni, 2010)

Malathy et al. used statistical regression models to predict the color of leather from its wet state but used a root-mean-squared error criterion. (Jawahar et al., 2013) However, a more commonly accepted criterion for color differences is either ΔE_{2000} or ΔE_{CMC}. (Melgosa et al., 2017; Oulton & Westland, 2017) In 2014, Malathy compared a neural network model with a Kubelka-Munk model to predict the final color of leather with thee dye combinations. Using CMC as the color difference formula to judge the prediction accuracy, Malathy concluded that an ANN model could predict the final color of leather with a DECMC value of 0.78 compared to a 2.65 for the Kubelka-Munk model. (Jawahar et al., 2015)

1.7. Neural Network

Neural networks, also named as artificial neural networks (ANNs) or simulated neural networks (SNNs), are a series of algorithms that mimic the operations of a human brain to recognize relationships related to vast amounts of data. Their name and structure are inspired by the human brain, mimicking the way that biological neurons signal to other neurons. A neural network model consists of 3 parts: an input layer, a hidden layer, and an output layer. The input layer is the beginning part of a neural network model and represents the input data. These can be as simple as scalars or as complex as vectors or multidimensional matrices. Relative to the input layer, the output layer is the final part of a neural network model. The output layer is used to produce the predicted values from the models. The layer between the input and output layer is named the hidden layer. The hidden layer consists of one or more layers with multiple neurons/nodes in each layer. A neuron is the processing unit of the network. Each neuron weights and sums the different inputs with the bias and passes them through an activation function. The role of the activation function (also known as the transfer function) is to buffer the

data before it is fed to the next layer. (Grossberg, 1988) The activation function is used to determine the output of an individual neuron. It maps input values to output values ranging from 0 to 1 or from -1 to 1. The activation function can be generally divided into two types: linear activation functions and non-linear activation functions. A linear activation function is just like the equation $f(x) = ax$ with a range from negative infinity to positive infinity. Non-linear activation functions are the most widely used activation functions for neural network models because they are easier for the model to generalize or adapt with a variety of data and to differentiate between the outputs. The non-linear activation functions are mainly classified by their ranges or their curves. The sigmoid/logistic activation function is the most common non-linear activation function and has an S-shape curve with a range from 0 to 1. It is especially well suited for models to predict the probability as an output. (Han & Moraga, 2005; Harrington, 1993) Hyperbolic tangent activation function is also a logistic sigmoid function but with a range from -1 to 1. The advantage of the hyperbolic tangent (tanh) activation function is that the negative inputs will be mapped strongly negative, and the zero inputs will be mapped near zero in the tanh graph. The tanh function is mainly used for classification between two classes. (Che-Wei Lin & Jeen-Shing Wang, 2008) Another activation function is rectified linear unit activation function (ReLU), and it is applied in almost all convolutional neural networks and deep learning. (M. M. Lau & K. Hann Lim, 2018; Olimov et al., 2021) The shape of ReLU looks like an obtuse angle, with a range from 0 to infinity. It is half rectified (from the bottom). $f(z)$ is zero when z is less than zero and $f(z)$ is equal to z when z is equal to or greater than zero. (Jin et al., 2015) A disadvantage of ReLUs is that all the negative values are mapped to zero, which decreases the ability of the model to fit or train the data properly. One method of overcoming this is called the Leaky ReLU, which ranges from negative infinity to positive infinity. The leaky ReLU helps to

increase the range of the ReLU function by adding a weighting factor to the linear model, which is $f(z) = az$ when z is less than 0 and a is 0.01. If the value of a is not 0.01, the ReLU activation function is called the randomized ReLU function. (Dubey & Jain, 2019; S. Fujii & H. Hayashi, 2019; Zou et al., 2008)

Training a neural network can be viewed as a nonlinear optimization problem in which the goal is to find a set of network parameters minimizing a cost (or loss) function. In other words, training is an optimization processing that produces an output as close as possible to the desired output by adjusting the model's weights and biases. This kind of parameter estimation is called a learning or training algorithm. The process by which the algorithm adjusts its weights is through gradient descent, allowing the model to determine the direction to take to reduce errors (or minimize the cost function). With each training example, the parameters of the model adjust to converge at the minimum gradually. (Sayed & Baker, 2015)

Backpropagation (BP), short for 'backward propagation of errors', is an algorithm for supervised learning of artificial neural networks using gradient descent and is widely used for training the feedforward neural network. With a fixed artificial neural network and a fixed error function, the method calculates the gradient of the error function with respect to the neural network's weights. It is a generalization of the delta rule for perceptions to multilayer feedforward neural networks. The delta rule is a gradient descent learning rule for updating the weights of the inputs to artificial neurons in a single-layer neural network. (Grossberg, 1988; Russell, 2012; Wythoff, 1993)

The "backwards" part of the name stems from the fact that the calculation of the gradient proceeds backwards through the network, with the gradient of the final layer of weights being calculated first and the gradient of the first layer of weights being calculated last. Partial

computations of the gradient from one layer are reused in the computation of the gradient for the previous layer. This backwards flow of the error information allows for efficient computation of the gradient at each layer versus the naive approach of calculating the gradient of each layer separately. (HECHT-NIELSEN, 1992; Yves Chauvin & David E. Rumelhart, 2013) Backpropagation's popularity has experienced a recent resurgence given the widespread adoption of deep neural networks for image recognition and speech recognition. It is considered an efficient algorithm, and modern implementations take advantage of specialized GPUs to further improve performance. (H. Jang et al., 2008; Lopes & Ribeiro, 2009)

Chapter 2: Materials and Methods

2.1. Materials and Machines

The following materials were used in the research. Small woven cotton swatches were cut from an untreated cotton fabric roll. Reactive dyes were purchased from Huntsman Corporation. Three primary reactive dyes (red, blue, and yellow) were selected: Novacron Red FN-3GL, Novacron Blue FN-G, and Novacron Yellow FN-2R. Sodium carbonate and sodium sulfate were ordered from Tronox Chemicals Company and Brenntag Connecting Chemistry. The dyeing machine used was a Datacolor AHIBA ECO (**Fig 2.1.**). The padding machine was from Werner Mathis AG. The spectrophotometer was an X-rite Color i7 benchtop spectrophotometer operated by Color Icontrol software (**Fig 2.2.**).

Figure 2.1. Datacolor AHIBA ECO dyeing machine (left: operation panel; right: inside of machine)

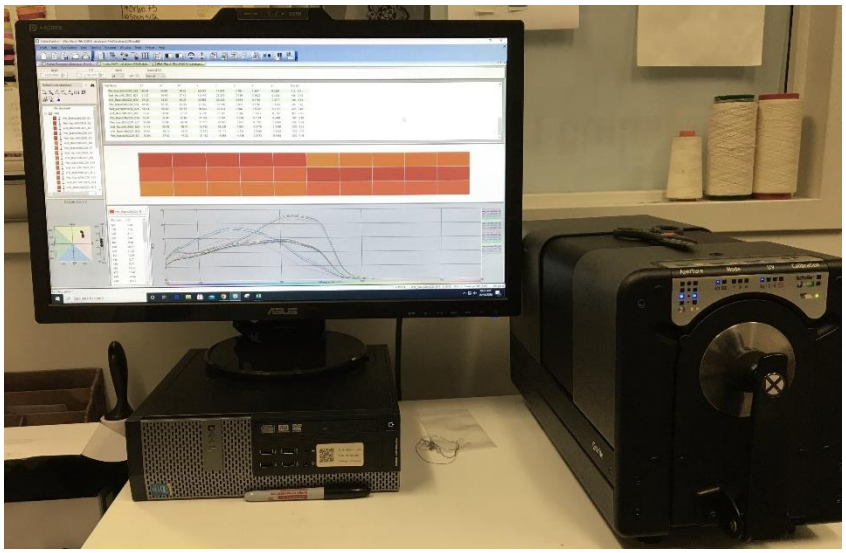


Figure 2.2. Experimental Design. X-rite Color i7 spectrophotometer and Icontrol software

Cotton fabrics were cut into small pieces for future dyeing and color measurements. The size of each cotton sample was 150 mm x 150 mm, and the mass of each small sample was around 1.50 g. The concentration of dye and other chemicals depended on the mass of each cotton sample. Dyes were diluted with DI water to make a 2.0 g/L stock solution for future dye baths. All the dye and chemical stock solutions were made daily and were mixed fully before using. The concentration of dyes was 0.1%, 0.2%, 0.25%, 0.5%, 1% and 2% on-weight-fabric (o.w.f.) for each primary color. The concentration of the sodium carbonate and sodium sulfate depended on the concentration of the total dyes, usually 15 g/L and 40 g/L, respectively (see **Table 2.1**). The dye bath ratio was 1:20 for each sample. (Reddy et al., 1997)

Table 2.1. Example of dyeing recipes

Data & Name	**July17th19_1**	**July17th19_8**	**Aug12th19_6**	**Aug12th19_13**
Padding pressure(bar)	2	2	2	2
Padding speeding(m/min)	2.0	2.0	2.0	2.0
dry weight(g)	3.31	3.18	2.9	2.81
Dry name	Novacorn Brilliant Red FN-3GL	Novacorn Brilliant Red FN-3GL	Novacorn Brilliant Blue FN-G	Novacorn Brilliant Blue FN-G
owf(%)	2	4	2	4
sodium sulfate (g)	5	7	5	7
soda ash (g)	1.6	2	1.6	2

2.2. Preparation for the dye bath

Cotton samples were weighed to three significant places, and the dyes and chemicals were calculated and measured following the specified formula for each dye recipe. Measured quantities of DI water, dye stock solution, sodium sulfate, and sodium carbonate were slowly added into the machine beaker and stirred slowly with a glass rod to make them dye evenly. The cotton samples were wetted with DI water and then processed through a padding machine to get rid of any excess water. The wet cotton samples were then placed into the dye bath beaker with a lid closed to ensure that no leaking or evaporation occurred.

2.3. Setting up the dyeing machine

All the setup beakers were placed in the dyeing machine, and their positions were recorded along with each beaker's dye formula. The dyeing procedure is given schematically in Fig 2, where time is the independent variable increasing from left to right.

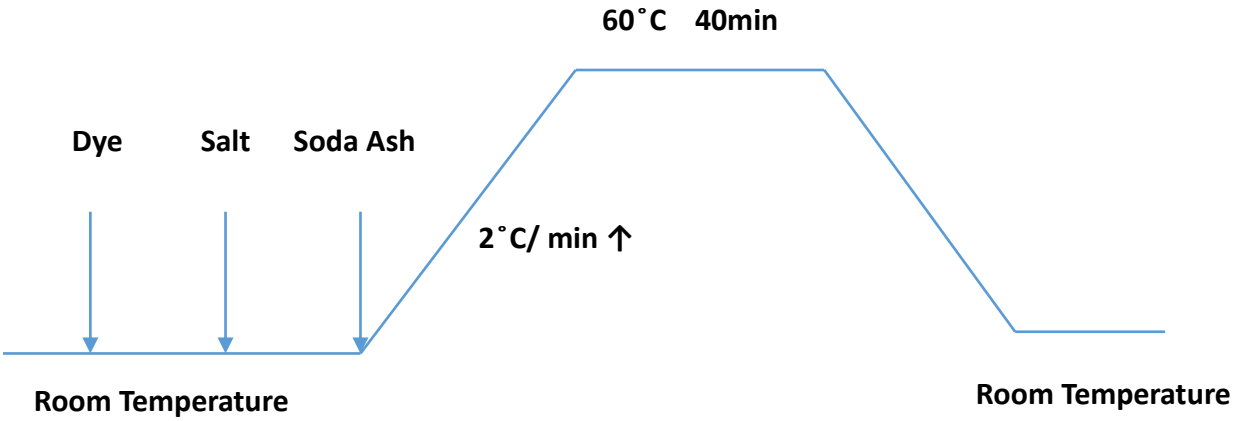


Figure 2.3. Dyeing Process

2.4. Color measurement and Data explanation

Every colored sample was wetted with DI water in a 2 L glass beaker for 10 seconds and passed through the padding machine with 2 m/min rolling speed and 0.5/1/2/4 bar pressure between two rollers. Then the fabric was rotated 90˚and then passed through the padding machine a second time with the same roller speed and pressure in order to finish the padding process. After padding, every wet sample was immediately measured to obtain the color in CIE L* a* b* units under the D65 light source and Large Area of Viewport (LAV, 25mm) using an X-rite spectrophotometer. The samples were then placed in an oven until completely dry. After drying in the oven, the samples were cooled for 10 minutes to room temperature. The dry color measurements were obtained under the same viewing conditions as the wet color measurements. Each set of dry and wet color measurements was combined together and recorded as one group for future data analysis. (See **Fig 2.4.** and **Table 2.2.** as examples)

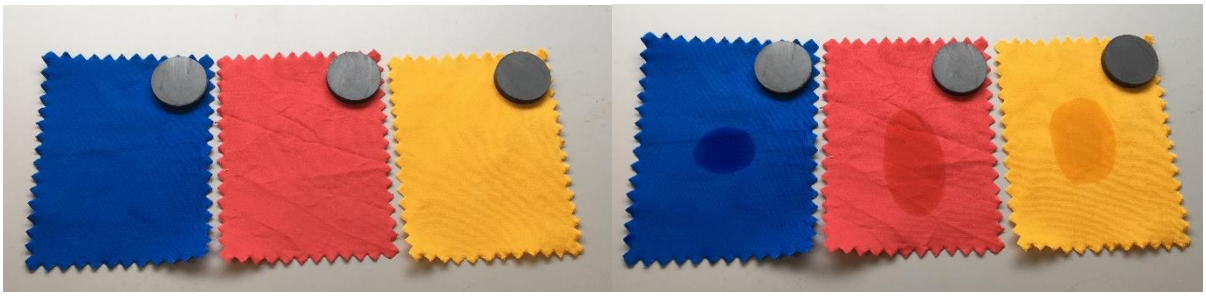


Figure 2.4. Pictures of dyed samples under D65 light source & observed at 45° (left: samples dyed by 0.5% o.w.f. Novacron Red FN-3GL, Blue FN-G, and Yellow FN-2R; right: samples with one drop of DI water.)

Table 2.2. Example of training datasets

Sample #	Pressure(Bar)	Dry			Wet		
		L*	a*	b*	L*	a*	b*
092419B1_1	0.5	69.45	41.88	39.36	63.28	49.05	64.85
021121B2_10	0.5	49.12	-3.18	-28.52	32.67	1.85	-30.86
020421B2_3	0.5	56.54	54.33	39.15	46.73	60.82	46.11
092419B1_10	1	66.41	48.64	47.73	61.45	55.05	69.39
092419B1_1	1	69.45	41.88	39.36	63.78	49.43	65.02
100319B_6	1	50.54	-6.77	-16.04	28.06	-3.12	-16
021121A1_1	2	55.65	-4	-26.25	41.66	-1.04	-31.34
021121A2_10	2	65.29	-4.28	-22.29	50.84	-3.72	-30.47
100319B_2	2	59.83	-6.16	-14.4	40.07	-5.83	-18.97
081519_1	4	51.51	-7.56	-38.81	38.27	-3	-43.11
081519_16	4	36.56	0.16	-42.28	23.63	5.74	-36.79
080819_9	4	44.4	52.87	18.95	38.93	58.37	35.92

Wet pick-up is defined as the percent of water remaining in a cotton fabric after squeezing out the water by two padding rollers. The wet pick-up of the fabrics is represented by the pressure set at the padding machine. A pressure of 0.5 bar, 1 bar, 2 bar, and 4 bar corresponds to wet pick-up of 110%, 87%, 76%, and 66%, respectively. The pressure was initially treated as an input parameter along with the value of L* a* b* in the wet state to train the neural network, but later removed, and separate models were constructed at each pressure level.

2.5. Timeline of the project

This project began with a literature review and a design of experiment, which lasted about two months, followed by dyeing cotton fabrics and color measurements. For each eight color samples, it was estimated that it would take 2 hours for experiment preparing, dyeing, and rinsing, and another 2 hours for drying, padding, and color measurement. In total, it took 4 hours to collect eight different color samples that were eventually used in training and testing the neural network. Since 765 different color samples were dyed and measured, it took more than 380 hours to create the training and testing color database for modeling. In total, the dyeing and data collection portion lasted about one and half years.

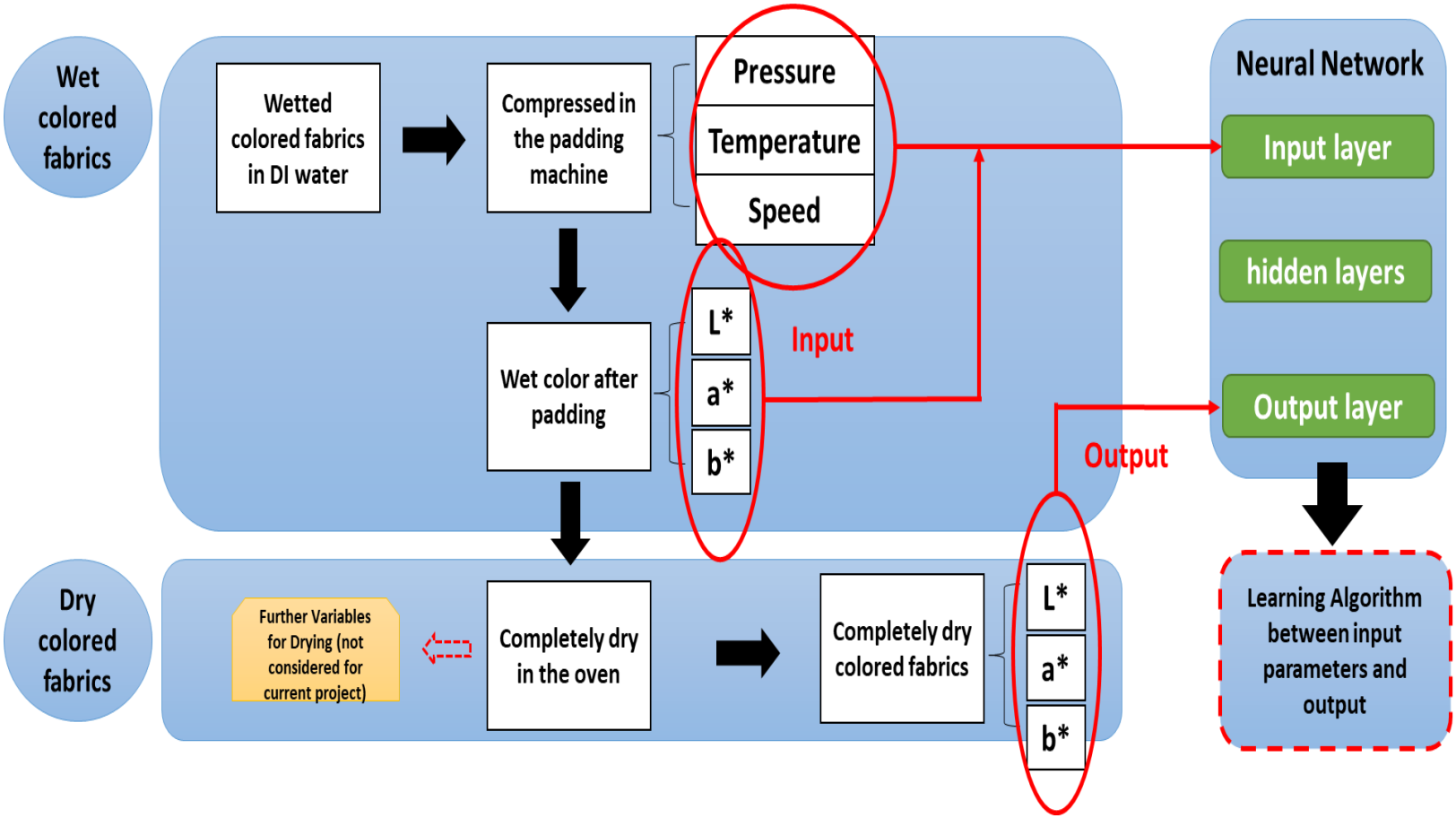
When the sample size of the color sample database exceeded 450, the first neural network models for wet and dry color prediction were attempted. Models were built and tested incorporating additional information from the literature, optimizing modeling promotion and parameter selection for higher prediction accuracy. After a few iterations, the final neural network models for predicting the color of cotton fabrics in the dry state from their color in the wet state were completed with a more than 90% prediction accuracy. At the beginning of the

modeling period, because of lack of modeling experience, it took hundreds of hours to train the models, and the attempted models did not converge. As recorded, it took around 1200 hours to generate the prototype of the prediction model. After completing the prototype of the prediction model, parameter tuning and model optimization were started with the help of NCSU's high-performance computers (HPC). Usually, the core used for the HPC was set at 24, and the batch script of the shell file for HPC is explained in the appendix. It took about another 400 CPU hours to get the final prediction models for the whole project. In total, it took over 400 hours to create the database and another 1600 hours to build the neural network prediction model. It is a 2000-hour project to develop neural network models for predicting the color of the dry cotton fabrics from their color in wet state.

2.6. Flowchart of the project

Figure 2.5 shows the flowchart of the whole project. The data was separated into two parts: dry data and wet data. The L* a* b* value of fabric's color in the dry state were trained as the output for the model. The L* a* b* value of fabric's color in the wet state along with machine and environment parameters such as the pressure of the compressed rollers, the speed of the compressed rollers, and the room temperature were trained as the input for the model. The speed of the compressed rollers and the temperature of the room were set as control. The speed was set at 2.0 m/min at a room temperature of 20 ° C. The pressure of the compressed rollers represented to the wet pick-up level was trained as a classification input explained later.

Figure 2.5. Flowchart of the project



2.6. Data Statistics and Data cleaning

A total of 765 dyed samples were used to train and test each model in this project, of which 37% of the samples were dyed with a single primary dye. Another 40% of the training samples were two-dye combinations, and the remaining 23% of the samples were three-dye combinations. (Figure 2.6.) The color of each sample was read and recorded under dry conditions or four different wet conditions (compressed under 0.5/1/2/4 bar). With these concentrations of the dyes, the 765 colored samples covered the available color gamut shown in figure 2.7. In Figure 2.7, x-axis is the x value of the color, and y-axis is the y value of color. The black points represent each color sample located in the xy surface.

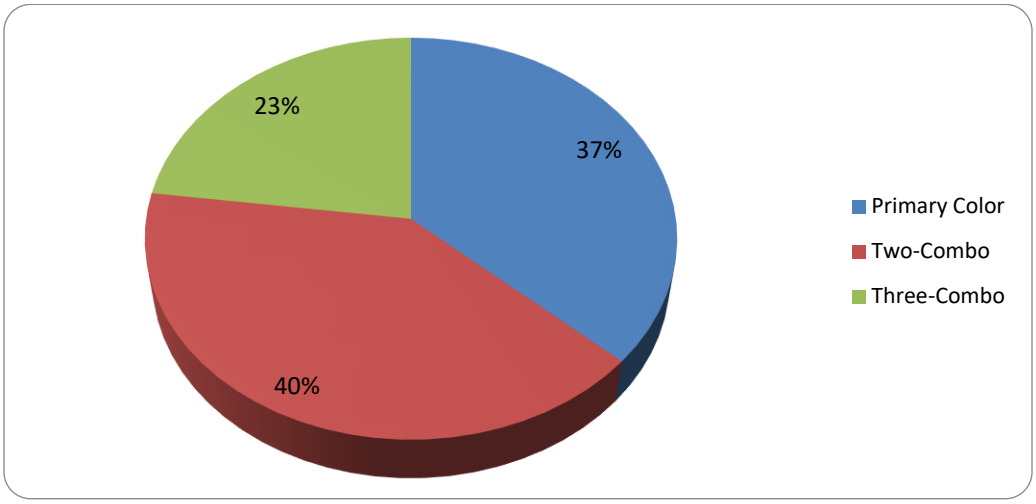


Figure 2.6. Distribution of Training Samples

Figure 2.8 shows 3D plots for 765 color samples in the wet and dry states located in L*a*b* color space. The range of L* is from 0 to 100 and the range of a* and b * is from -100 to 100. All the points are colored depending on their value of L*. Obviously, the shape of 3D plot is not a completely perfect sphere because of the limitation of three primary dyes, but it covers most of the achievable color gamut. The L* a* b* range of 765 color samples in the dry state is shown in table 2.3.

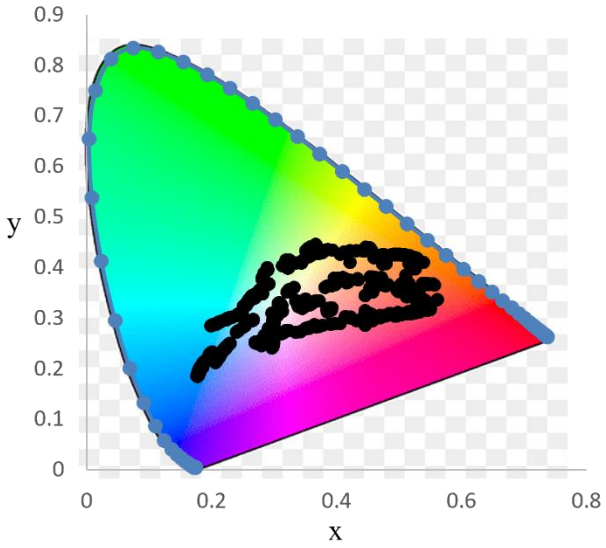
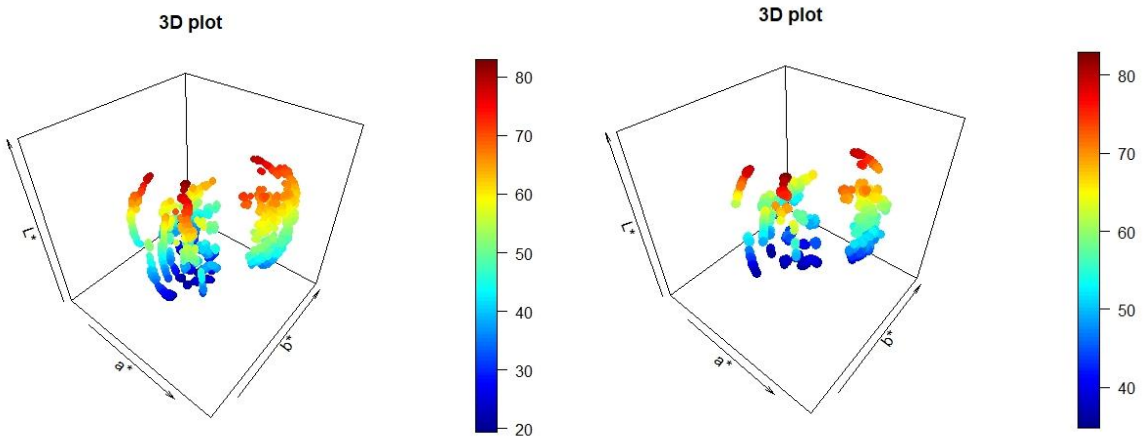


Figure 2.7. Color gamut for 765 samples in xy plot



(a) (b)

Figure 2.8. 3D plot for 765 color samples in wet (a) and dry (b) state located at L*a*b* color space.

After data collection, all the dyed samples have five sets of L*a*b* color values (dry color, wet color under 0.5 bar, wet color under 1 bar, wet color under 2 bar, and wet color under 4 bar). All of the color values (L*, a*, and b*) were normalized in the range of 0 to 1 using Equation 1.

$$y_i^* = \frac{y_i - \min(y)}{\max(y) - \min(y)}, \; i = 1, 2, 3, \ldots, \mathrm{n} \qquad (2.1)$$

where $y = (y_1, y_2, \ldots, y_n)$, y_i^* is the normalized value, y_i is the measured color (L*, a* or b*), and n is the total number of samples. The R code for normalizing the raw data is shown in Appendix.

Table 2.3. Range of L*a*b* of 765 color samples in dry state

Range	Minimum	Maximum
L*	19.47	83.90
a*	-47.42	62.77
b*	-45.38	85.29

2.7. Modelling

First, the 765*4 (each color under four different pressures) wet color values were randomly distributed into the training and testing group together with their related dry color values. All the normalized datasets were separated randomly into a training dataset (80% of the total dataset) and a testing dataset (20% of the total dataset) for modeling. Then, based on

different squeeze roll pressures, wet color values were separated into four different groups, for which there are 765 datasets in each group. For each group, the normalized datasets were distributed randomly into a training dataset (80% of the total dataset) and a testing dataset (20% of the total dataset) for modeling. In other words, the first method used pressure as one of the input parameters along with L* a* b* values in the wet state for modeling, and the second method pre-classified the data into four groups by their pressure and then generated four different prediction models for each pressure.

Neuralnet is a neural network package written in the R programming language used to build and test the prediction models. (Günther & Fritsch, 2010) Training of neural networks was accomplished using backpropagation. (Gu et al., 2018; Riedmiller, 1994) The L*, a*, b* values in the wet state under different pressures were used as input data, and the L*, a*, b* values in the dry state were used as output data. The pressure of the rollers was used as another input data for the first model. The number of the hidden layers and the number of the nodes in each hidden layer were modified and selected by the final prediction results and compared to the real data. Parameter tuning for the learning thresholds step-max were also optimized. The CIE_{76} (the Euclidian norm, the square root of the sum of the square of ΔL, Δa, and Δb), CIE_{96}, ΔE_{2000} and ΔE_{CMC} color difference formula were used as the error functions for training. (HECHT-NIELSEN, 1992; Heggie et al., 1996; Luo et al., 2001; McDonald & Smith, 2008; SÈVE, R., 1996; Sève, Robert, 1991; Sharma et al., 2005) The training dataset (comprising 612 samples) under different pressures was used to train the model, and the testing dataset (containing 153 samples) was used to test the accuracy of the model. The first model with the pressure of the rollers as one of the input data could not be converged no matter what parameter setting for the models, therefore, we used the pressure of the rollers as categorial data to divide all the data into

four groups and created four different models by each pressure separately. For the testing results, we used ΔE_{2000} color difference formula as the criterion to calculate the final color difference between the predicted dry color from each model to the real dry color. Commonly color difference $\Delta E \leq 1$ is not able to recognize as different colors by human eyes, and $1 < \Delta E \leq 2$ between two colors can be noticed as color difference by close observation. But for the color difference in textiles, we used $\Delta E \leq 0.7$ as the tolerance to judge the prediction accuracy for the model in textiles. (Brainard, 2003) Appendix 3 shows an example of an R script on how to build a neural network for prediction from wet color to dry color.

Chapter 3: Discussion

3.1. Structure of the Neural Network Model

The squeeze-roll pressure was initially treated as an input parameter along with the values of L* a* b* in the wet state to train the neural network. Various methods were attempted, such as different levels of thresholds, step-max, and the number of nodes in the hidden layer. However, the model training section took over 48 hours to run, and the final models did not converge. Therefore, four separate models were developed by pre-classifying the data based on the squeeze pressure. Figure 3.1 shows the structure of the final model as an example. Different squeeze roll pressures, corresponding to different wet pickups, required different ANN topologies as shown in Figure 3.1.

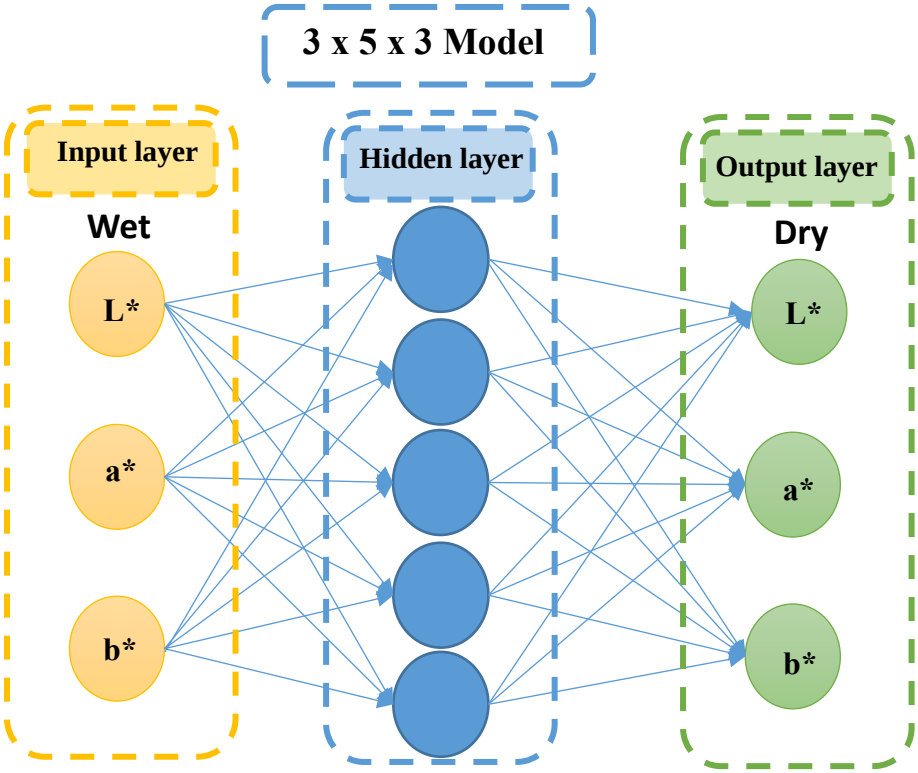


Figure 3.1. The structure of 3*5*3 Neural Network model

The L*, a*, b* values in the wet state under different pressures were used as input data and the L*, a*, b* values in the dry state were used as output results. The number of the hidden

layers and the number of the nodes in each hidden layer were modified and selected by the final prediction results compared with the real data. For example, in Figure 3.1, a 3*5*3 model represents L*, a*, b* values in the wet state as three nodes in the input layer, five nodes in the hidden layer, and L*, a*, b* values in a dry state as three nodes in the output layer.

3.2. Models with various error functions

As mentioned before, ΔE_{76}, ΔE_{94}, ΔE_{CMC}, and ΔE_{2000} color difference formulas were used as the error function in backpropagation training for comparison to determine the optimal metric to use to both train and evaluate the models. Therefore, we used the same datasets to train and test a 3*8*3 model using four different color difference formulas and then compared them. The ΔE_{2000} values between the predicted dry color from the four trained models and the real dry color are shown in Figure 3.2 and Table 3.1. In Figure 3.2, X axis means the marked number of the testing samples ranked from lowest ΔE_{2000} color difference value to the highest ΔE_{2000} color difference value, and Y axis means the value of ΔE_{2000} color difference. It is obvious that ΔE_{2000} color difference model (blue spots) has the lowest color difference compared to other three models in Figure 3.2.

In Table 3.1, each column corresponds to a given model that was trained and tested with a particular error function, and then those predictions were evaluated against the four different error functions (one for each row). For example, in the first row, the ANN was evaluated using the ΔE_{2000}, and the mean error was calculated for each model. Notice that regardless of the error function used to evaluate the models, the ΔE_{2000} model always gave the lowest value (the first column). In other words, for each row, the value in the first column was the lowest. Therefore,

we concluded models trained and tested using the ΔE_{2000} error function gave the best results (without bias).

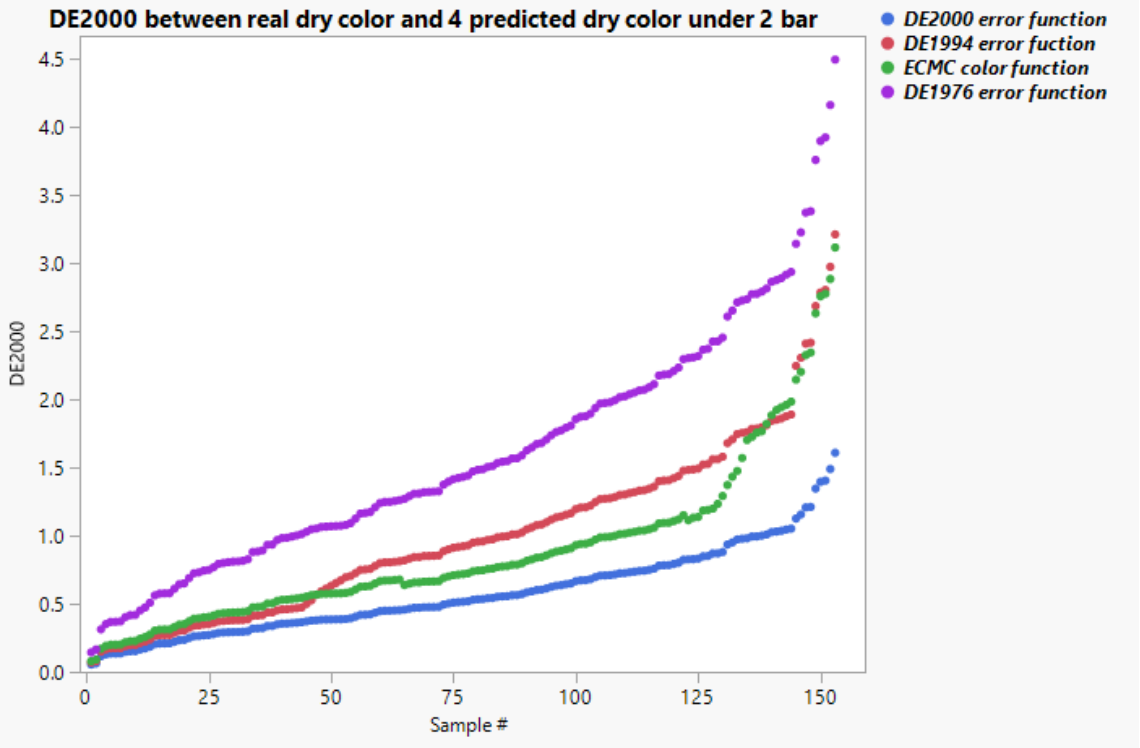


Figure 3.2. ΔE_{2000} between the dry and predicted dry color from 4 models with different error functions for 3*8*3 structure under 2 bar pressure

Table 3.1. Color difference for models with 4 error functions

	ΔE_{2000} Model	ΔE_{1976} Model	ΔE_{1994} Model	ΔE_{CMC} Model
ΔE_{2000} Mean	0.58	1.58	0.99	0.88
ΔE_{76} Mean	0.75	2.3	1.50	0.98
ΔE_{94} Mean	0.50	1.51	0.99	0.67
ΔE_{CMC} Mean	0.79	2.33	1.53	1.05

3.3. Models with different structures

Once the optimal error function was determined, the structure of the model was then optimized for each squeeze roll pressure. Initially, we generated a different model for each output. For example, a 3x3x1 * 3 model is a set of 3 models, one for each output. Each model is used to predict the color of L*, a*, and b* separately, and there are three nodes in the hidden layers. 3x4x1 * 3 and 3x5x1 * 3 have a similar structure. The 3x5x5x3 model is another different model, which has two hidden layers with five nodes in each hidden layer. The results of model prediction are shown in Figure 3.3. Three sets of models (3x3x1 * 3, 3x4x1 * 3, and 3x5x1 * 3) didn't show good prediction results, as over 80% of the predicted data had an ΔE_{2000} color difference greater than 1.0, which is beyond the tolerance. Therefore, a structure of a set of 3 models was not a good prediction model. Since training time is another consideration, neither the model with two hidden layers nor the model with more than ten nodes in the single hidden layer trained the model in 48 hours. As an example, it took seven days to train the 3x5x5x3 model. Additionally, the prediction accuracy of the model with two hidden layers and the model with more than ten nodes in the single hidden layer is not as good as that from the models with less than ten nodes in the single hidden layer. Therefore, we found that a 3*x*3 (x is in the range from 6 to 9) structure produced the best models at different pressures.

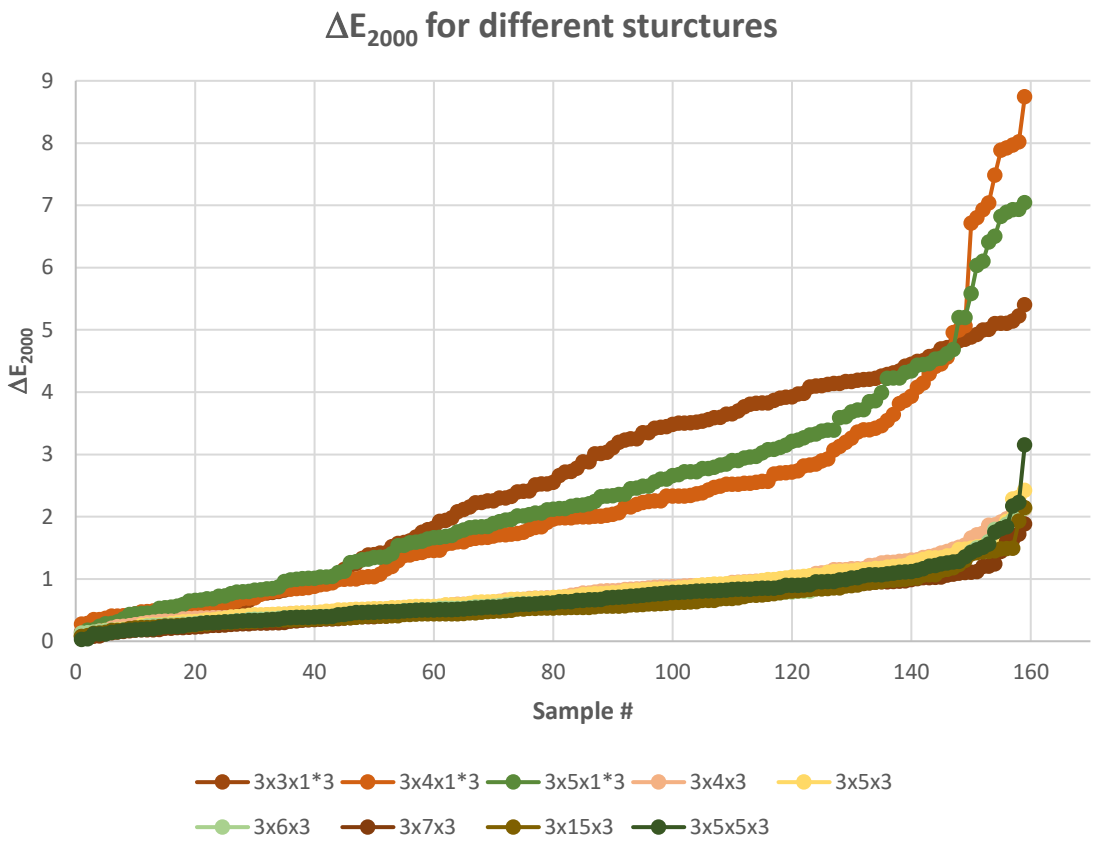


Figure 3.3. ΔE_{2000} for different structures of the models

3.4. Parameter tuning for the model

A few parameters need to be optimized or tuned for the models to converge and give reasonable predictions. The error functions were discussed in the previous section, and it was concluded that ΔE2000 is the best method to calculate and estimate the bias between the prediction output and the actual values. Additionally, the tolerance value for the prediction to pass or fail the test was the next question that needed to be considered. In an artificial neural network, a neuron is activated by a weighted signal from the previous neurons. There are also different conditions by which to judge the results as a pass or fail. Those conditions are called

thresholds in neural network models. The threshold is used to generate the signal and transfer predicted values to the next neurons or terminate this learning loop and start the next loop again. For example, Fig 3.4 shows a simple part of a neural network model with two input nodes and one output node. F(x) can be given the simple formula as $F(x) = x_1+x_2$, and the threshold here is 'no more than 20'. Assume, for example, that the input values for x_1 and x_2 are 5. The result of F(x) is smaller than 20, which means when the value of x_1 and x_2 is 5, the prediction model passes the test and generates output to the next neuron or prints out that result.

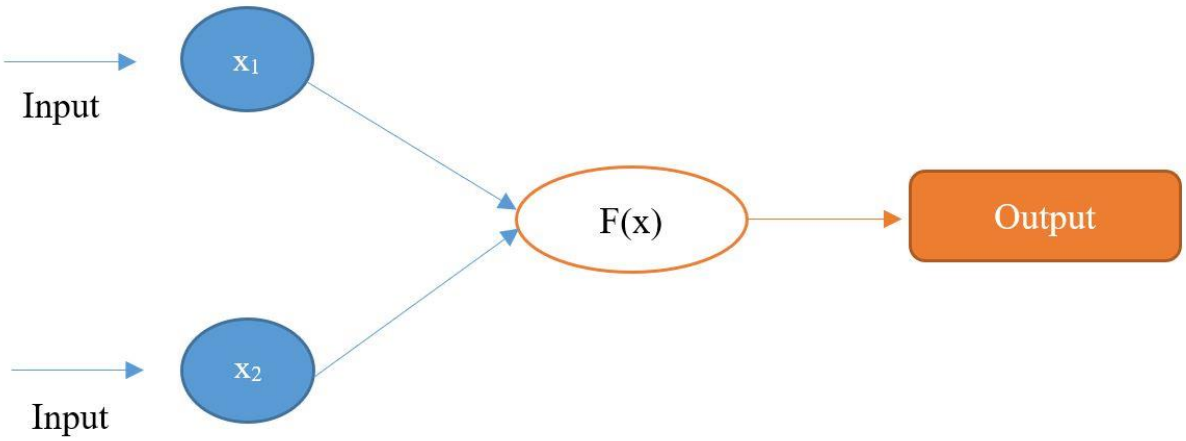


Figure 3.4 Example of 2 * 1 model

In our project, the selection of the threshold for the models directly affects the prediction accuracy for the value of L*, a*, and b* for the predicted dry color. If the threshold were set too high, the prediction accuracy for the output would decrease, and the final model would have a poor performance for predicting the color. If the threshold were set too low, it would cost a lot of time to let the model learn, and the model may not converge. In Table 3.2, we used the same dataset to train the 3*8*3 models with ΔE_{2000} as the error function and the wet color measured under 2 bar pressure. The only difference is the setting of the threshold value. From this table, if the threshold value setting is too small (less than 0.01), the model does not converge. If the

threshold value is set too large, the prediction accuracy decreases sharply. Therefore, based on the training time efficiency and model prediction accuracy, the best threshold value is 0.01 for this project.

Table 3.2. Neural network models under 2 bar pressure with different threshold

Threshold	Model convergence	Train time	Prediction accuracy
0.02	Converged	~ 48 hours	75.66%
0.01	Converged	~ 48 hours	90.20%
0.005	Not converged	Over 72 hours	N/A
0.001	Not converged	Over 120 hours	N/A

Stepmax is another parameter needed by the Neuralnet package**.** *Stepmax* represents the maximum number of steps or epochs for training the neural network, and reaching this maximum leads to a stop of the neural network's training process. If the model reaches the maximum number of epochs without converging during the training process, the program exits. If *stepmax* is set too big, the model may fail to converge and waste CPU cycles. If the *stepmax* parameter is set too small, the model may not have enough chances or steps to predict accurately. For example, we used the same dataset to train the 3*8*3 models with ΔE_{2000} as the error function and the wet color measured under 2 bar pressure. The parameter tuning in Table 3.3 is the *stepmax* for the same model during the training process. When the *stepmax* parameter is set too big (1e+10, 1e+11, or higher), the model fails to converge. When the step-max parameter was selected too small (1e+06 or smaller), the model did not have enough steps or epochs to learn, and the final model did not predict well. After considerable trial and error, it was determined that a value of 1e+08 (100000000) is the best selection for the *stepmax* parameter.

Table 3.3. Neural network models under 2 bar pressure with different stepmax

Stepmax	Model convergence	Train time	Prediction accuracy
1e+06	Not converged	13 hours	N/A
1e+07	Converged	~ 48 hours	75.66%
1e+08	Converged	~ 48 hours	90.20%
1e+09	Converged	Over 72 hours	90.12%
1e+10	Not converged	Over 120 hours	N/A
1e+11	Not converged	Over 120 hours	N/A
1e+12	Not converged	Over 120 hours	N/A
1e+13	Not converged	Over 120 hours	N/A

3.5. Best prediction model for wet color under 4 bar pressure

Once the optimal structure of the model and error function were determined, prediction models for four different wet pick-up levels were trained under different roller pressure conditions. Starting with the dataset under a 4 bar pressure, 612 sets of L*a*b* values were used to train the model, and the remaining153 sets were used to test the model. *Stepmax*, a parameter that sets the maximum number of epochs or iterations, was set to $10^{10,}$ and the maximum total mean square error threshold was set to 0.01 during parameter tuning. Predicted errors for the four different models under 4 bar pressure are shown in Figure 3.5, X axis means the marked number of the testing samples ranked from lowest ΔE_{2000} color difference value to the highest ΔE_{2000} color difference value and Y axis means the value of ΔE_{2000} color difference. The 3*8*3 model gave the lowest mean (0.50) and smallest standard error (0.02) compared to the other three models at a 4 bar pressure (3*6*3 model with 0.74 mean and 0.03 standard error, 3*7*3

model with 0.73 mean and 0.03 standard error, 3*9*3 model with 0.70 mean and 0.03 standard error). In Figure 3.5, it is obvious that the 3*8*3 model (green spots) has a lowest curve compared to the other three models. Table A2 summarizes the results of 4 models under 4 bar pressure. The 3*8*3 model is the best model with 66% wet pick-up to predict the color in the dry state, where the maximum ΔE_{2000} error was reduced to 1.49. Only about 6.54% of total testing data has a color difference over 1.0, and about 24% of the testing data had an ΔE_{2000} over 0.7. Over 93% of testing data passed the color difference tolerance at 1.0, and over 75% of the testing samples could not be judged as even having a perceived color difference.

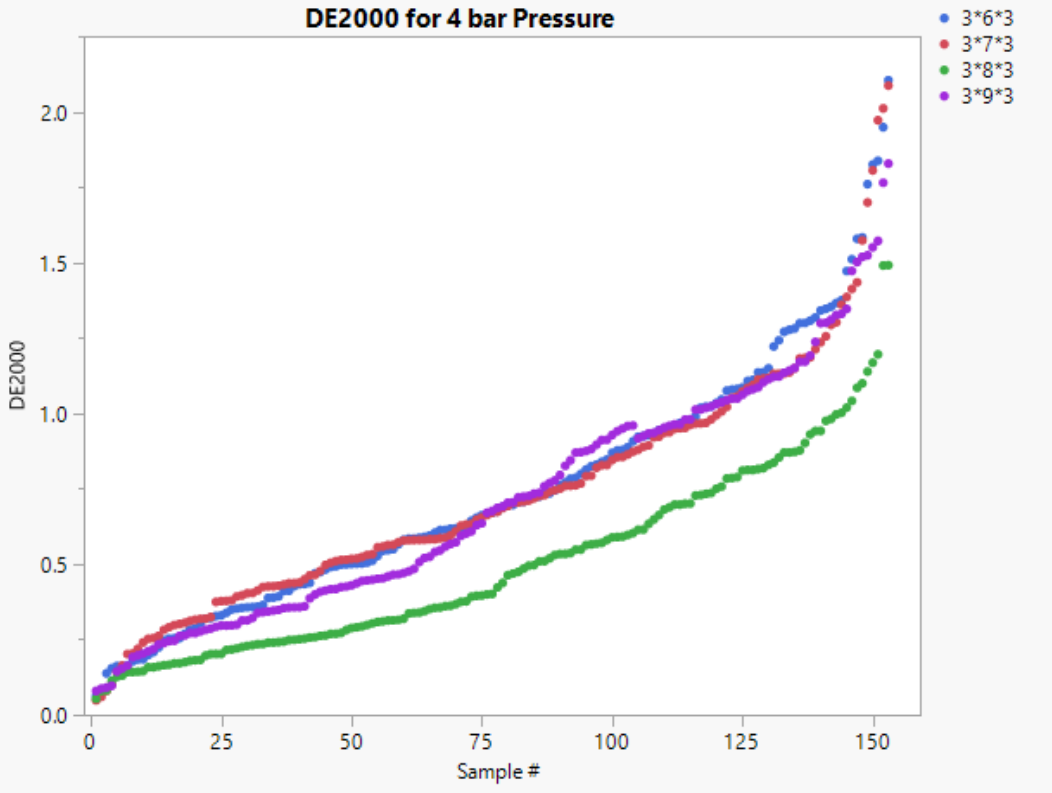


Figure 3.5. ΔE_{2000} of 4 models under 4 bar pressure

3.6. Best prediction model for wet color under 0.5/1/2 bar pressure

Similar to the 4 bar pressure models, 612 sets of L*a*b* values for the wet color under 2 bar pressure and related dry color data were used as inputs and outputs to train the model, and the remaining 153 data sets were used to test for model accuracy. The error for four different models under a 2 bar pressure is shown in Figure 3.6, X axis means the marked number of the testing samples ranked from lowest ΔE_{2000} color difference value to the highest ΔE_{2000} color difference value and Y axis means the value of ΔE_{2000} color difference. Although the 3*9*3 model shows similar results in the first half of the datasets, the 3*8*3 model still gave the lowest mean (0.57) and smallest standard error (0.03) compared to the other three models at a 2 bar pressure (3*6*3 model with 0.74 mean and 0.03 standard error, 3*7*3 model with 0.72 mean and 0.03 standard error, 3*9*3 model with 0.66 mean and 0.03 standard error). In Figure 3.6, 3*8*3 model (red spots) shows the relatively lowest color difference compared to others. Table A3 summarizes the results of the four models under 2 bar pressure. The 3*8*3 model is the best model from the wet color with 76% wet pick-up under 2 bar compressed pressure to predict the color in the dry state. The maximum color difference is reduced to 1.60. Only about 9.80% of total testing data has an ΔE_{2000} over 1.0, and about 31.4% of testing data has an ΔE_{2000} over 0.7. Similar results were obtained at 1.0 bar, and 0.5 bar and are shown in Figure 3.7- 3.8 and Appendix 1. 3*9*3 model (Purple spots) shows the lowest curve in Figure 3.7 for 87% wet pick-up under 1 bar compressed pressure to predict the color in the dry state and 3*6*3 model (blue spots) shows the smallest color differences compared to other three models in Figure 3.8 for 110% wet pick-up under 0.5 bar compressed pressure to predict the color in the dry state.

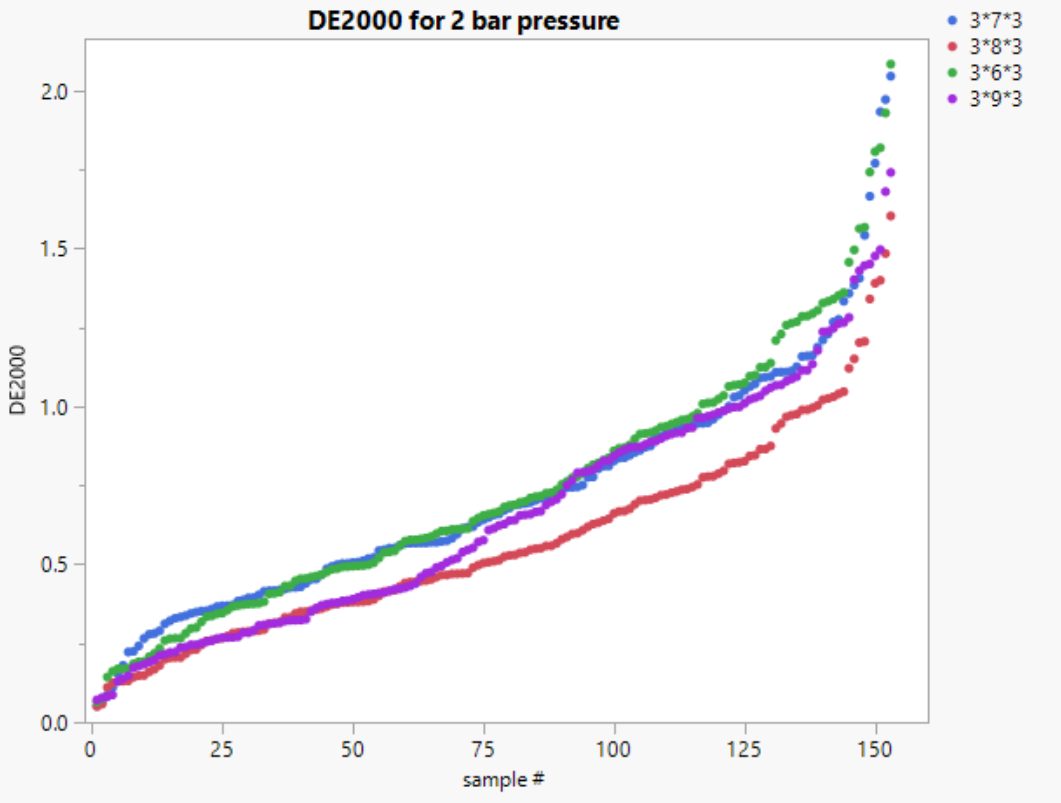


Figure 3.6. ΔE_{2000} of 4 models under 2 bar pressure

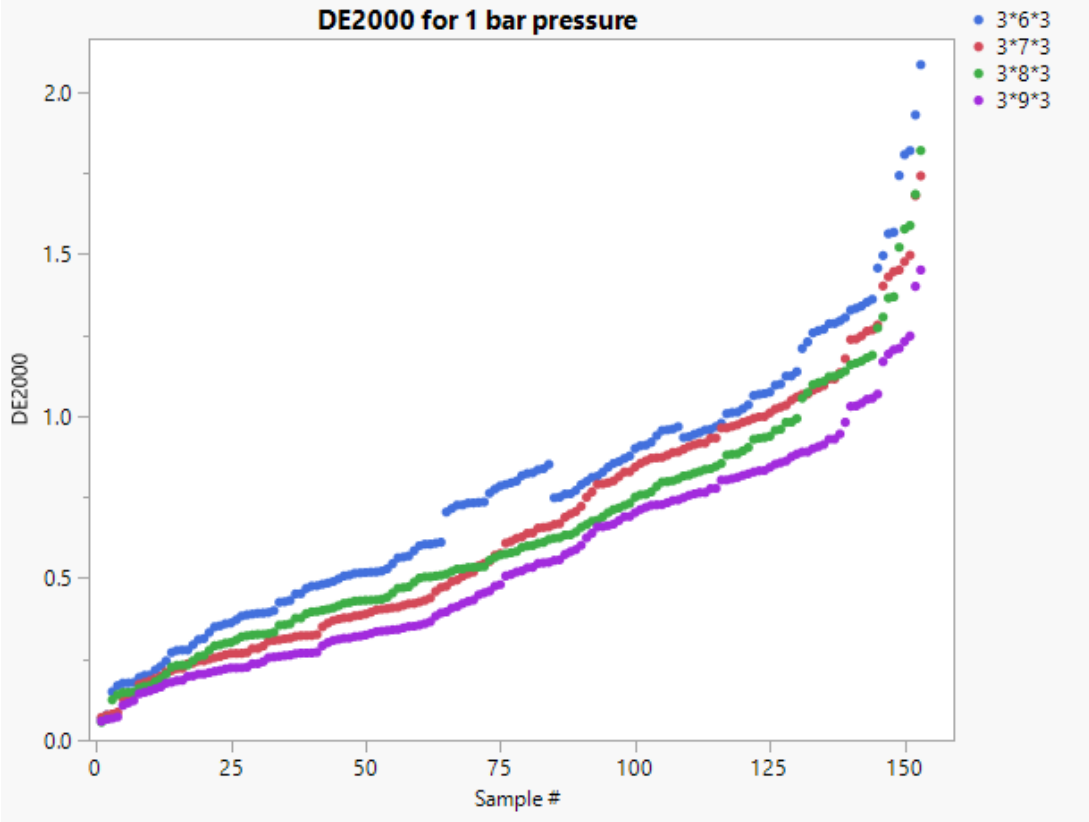


Figure 3.7. DE2000 of 4 models under 1 bar pressure

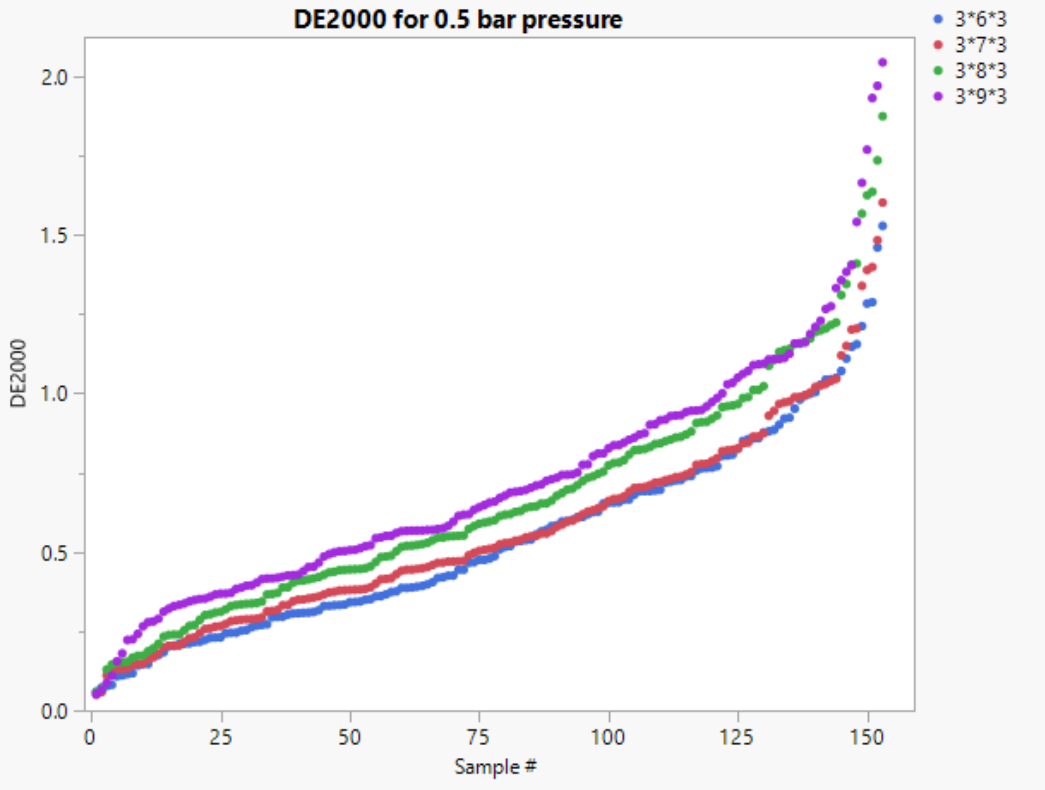


Figure 3.8. ΔE_{2000} of 4 models under 0.5 bar pressure

Chapter 4: Conclusion and Future work

4.1. Conclusion

Although water is a colorless liquid, it does cause a sizeable apparent color difference when applied to a dry fabric. Water is either absorbed into the fibers or is present between the fibers in the fabric which leads to a perceived color difference. This can be partially explained as a function of wet pickup, although different initial shades correspond to different color differences with the same amount wet pickup. Since models incorporating wet pick-up rates as one of the input parameters failed to converge (as discussed in in Chapter 2), the wet pick-up rates could not be used as a numeric input to train the model. Because of that, the effect of wet pick-up rates on color differences cannot be predicted by a linear regression model.

Neural network machine learning algorithms are able to create prediction models to predict the color of the cotton fabrics in a dry state from their wet color. By using the CIE_{96}, ΔE_{2000} and ΔE_{CMC} color difference formulas instead of a mean square error or CIE_{76} (the Euclidian norm, the square root of the sum of the square of ΔL, Δa, and Δb) as the error function, the neural network could better predict the color of fabrics in the dry state from measured L*a*b* values in the wet state. Comparing these models with four different color difference formulas, the models with ΔE_{2000} color difference formula as the error function showed the best prediction accuracy and lowest predicted color difference error mean. The outputs from ΔE_{2000} prediction models have much smaller color differences between the predicted L*a*b* values and their real measured L*a*b* values in the dry state, compared to the other small color difference formula models. The choice of the error function in the neural network model is a very important tool to affect the final prediction accuracy as well as the rates of convergence.

In total, 765 different color samples were dyed in this project: 80% were used for training and the remaining 20% were used for testing. Since all the color samples have unique L*a*b* values covering the achievable color gamut, the color samples could be randomly divided into training or testing groups. A detailed description of the different structures of the neural network models are described in Chapter 3. In consideration of the prediction accuracy and time efficiency, models with one hidden layer and 5 to 9 hidden nodes are the best choices for the structure of the models for this problem.

With an optimized model, over 90% of the predicted values were under a CIE ΔE_{2000} color difference of 1.0, which is generally considered within acceptable tolerances. The maximum color difference was successfully reduced from over 11 to less than 1.7, and the mean color difference was also sharply reduced from over 8 to under 0.6. Different wet pick-up rates corresponding to different squeeze roll pressures require different neural network structures (the number of hidden layers). The best neural network model for each wet pick-up rate is shown in Table 4.1. With a reliable model that works over such a wide range of values, one should be able to better control the dyeing process and achieve right-first-time results by being able to measure the fabric directly during the dyeing process as opposed to having to first dry the fabric sample or determine the final shade through indirect means.

Table 4.1. Best neural network model for each wet pick-up rates

Wet pick-up rates	**110%**	**87%**	**76%**	**66%**
Best model	**3*6*3**	**3*9*3**	**3*8*3**	**3*8*3**
Prediction accuracy	**90.9%**	**90.8%**	**90.2%**	**93.5%**
Maximum ΔE_{2000}	**1.53**	**1.45**	**1.60**	**1.49**

As a first attempt to combine color prediction and AI algorithms, the following is a methodology or framework to create a machine learning model to predict the color of the fabrics from their wet state:

1. Determine the input and output factors needed to be targeted, and assign the rest of the factors which may affect the final output as control factors. Select a set of dyes, a fabric, and a dyeing procedure.
2. For a given set of dyes, insure that the samples dyed cover the desired color gamut. Ensure a constant wet pickup for each data set.
3. During data cleaning, all the raw data should be normalized for modeling. Check for missing data and outlier points. Randomly separate all the data points into two groups, 80% for training and 20% for testing while ensuring good coverage of the achievable color space.
4. During modeling, optimize the best models by choosing the proper structures, and parameters of the neural network.
5. In the testing phase, characterize the quality of the model with the testing data. When needed, iterate on the design.

This methodology can be generalized to different dye and fabric types, and is not limited just to reactive dyes on woven cotton, but can be used on various substrates and dye classes. For example, direct dyes, acid dyes, and other series of reactive dyes can be used to generate their own prediction models and databases. In addition, with proper modifications, this methodology could be used in the finishing process such as coating and post-treatment. As the big data era is coming, generating digital databases would position the textile indust to incorporate AI machine

learning algorithms and statistics to current textile research could make advances to further unsolved problems in color science and dyeing.

The textile industry is a large consumer of energy and non eco-friendly chemicals. The ability to predict the color of fabrics in the dry state from the wet state is of practical significance for sustainability calculations. Energy for heating and processing contributes to greenhouse gases and global warming. The neural network prediction model will be able to save large amounts of energy, time, resources, and money which contributes to environmental protection.

4.2. Future Work

Wet pick-up rates corresponding to squeeze roller pressure were confirmed to be an important parameter affecting the model prediction. In our project, we treated roller pressure as a categorical parameter; however, with exact roller pressure control data, the roller pressure should be treated as a numerical parameter to train the neural network model. The speed of the roll is another factor that may or may not affect the wet pick-up rates of the dyed samples. The room temperature and humidity are also factors to be considered as variables to train the model and affect the color of the fabrics in the dry and wet states.

Since the model application and accuracy should increase with the size of the database, continually updating the database and model would yield more precision in model prediction. (Cui & Gong, 2018) Since only three primary reactive dyes were studied in this project, other dyes in the same series should be used to enlarge the color database and increase color gamut in the 3D space for training and higher model prediction accuracy.

Because of the poor prediction performance near the white and black areas regions, one of the limitations of the prediction model is to predict very light colors and very dark colors. According to Lin's research, there are many factors that affected the whiteness of fabric. (Lin, 2013) Another future area of research should focus on creating neural networks for white and black separately.

As the final goal is to generate a robust methodology for designing and implementing AI algorithms to build a color prediction model, the prediction model with different types of fiber (such as wool, nylon, blended fiber) and different types of dyes (such as disperse dyes, acid dyes) should be created.

Although the training model time is less than 48 hours with optimized parameters, simplifying the model computationally is another challenge. It is necessary to streamline the program coding and make the training session more time-effective. That should be introduced and combined with computer science, database construction, and basic computing to the current project.

CHAPTER 5: Data and Program Security

All of the raw data was collected and cleaned by Hanchi Zhu, supervised by Dr. Warren Jasper. The whole data and R program codes were stored in Dr. Warren Jasper's NCSU google drive shared with Hanchi Zhu. Some of the results in this project were selected for publishing a paper. The raw data is not given in the appendix. Anyone who wants to check or use the data about this project, please ask Dr. Warren Jasper for data access permission.

References

AATCC. (2015). *CMC: Calculation of Small Color Differences for Acceptability*

Allen, E. H., & Goldfinger, G. (1971). The Change in Color Of Textile Samples Upon Immersion in Water. *Textile Chemist & Colorist, 3*(12)

Allen, E. H., & Goldfinger, G. (1972). The color of absorbing scattering substrates. I. The color of fabrics. *Journal of Applied Polymer Science, 16*(11), 2973-2982.

Almodarresi, E. S. Y., Mokhtari, J., Almodarresi, S. M. T., Nouri, M., & Nateri, A. S. (2013). A scanner based neural network technique for color matching of dyed cotton with reactive dye. *Fibers and Polymers, 14*(7), 1196-1202.

Artificial Neural Network In Fabric Engineering. (2012, Nov 24,). *Textile Review,*

Athanasios A. Tsoutseos, & James H. Nobbs. (1999). Colour appearance of textile materials: an alternative approach. In Stephen M. Burkinshaw (Ed.), *Colour science '98: proceedings of the International conference & exhibition, Harrogate, April 1-3, 1998. Textile dyeing & printing, Volume 2* (pp. 234-246). Univeristy of Leeds.

Athanasios A. Tsoutseos, Nobbs, J., & Boussias, C. M. (1999). Methods of Improving the Colour Match Prediction in Textile Dyeing Using Novel Colour Appearance Models and Neural Networks. *3rd International Conference,* , 196-213.

Atherton, E. (1955). The Relation of the Reflectance of Dyed Fabrics to Dye Concentration and the Instrumental Approach to Colour Matching. *Journal of the Society of Dyers and Colourists, 71*(7), 389-398. 10.1111/j.1478-4408.1955.tb02089.x

Baghdadi, R., Alibi, H., Fayala, F., & Zeng, X. (2021). Investigation on Stiffness of Finished Stretch Plain Knitted Fabrics Using Fuzzy Decision Trees and Artificial Neural Networks. *Fibers and Polymers, 22*(2), 550-558. 10.1007/s12221-021-9314-8

Balci, O., Noyan Oğulata, S., Şahin, C., & Tuğrul Oğulata, R. (2008). An artificial neural network approach to prediction of the colorimetric values of the stripped cotton fabrics. *Fibers and Polymers, 9*(5), 604-614.

Bello-Cerezo, R., Bianconi, F., Fernández, A., González, E., & Di Maria, F. (2016). Experimental comparison of color spaces for material classification. *Journal of Electronic Imaging, 25*(6), 061406.

Berns, R. S. (2019). *Billmeyer and Saltzman's Principles of Color Technology*. John Wiley & Sons, Incorporated.

Brainard, D. H. (2003). 5 - Color Appearance and Color Difference Specification. *The Science of Color* (Second Edition ed., pp. 191-216). Elsevier B.V. 10.1016/B978-044451251-2/50006-4

Brinkworth, B. J. (1972). Interpretation of the kubelka-munk coefficients in reflection theory. *Applied Optics (2004), 11*(6), 1434-1435. 10.1364/AO.11.001434

Cárdenas, L. M. (2009). *Evaluation of variability in the assessment of small color differences* http://www.lib.ncsu.edu/theses/available/etd-02252009-141236/unrestricted/etd.pdf

Che-Wei Lin, & Jeen-Shing Wang. (2008). A digital circuit design of hyperbolic tangent sigmoid function for neural networks. Paper presented at the - *2008 IEEE International Symposium on Circuits and Systems (ISCAS),* 856-859. 10.1109/ISCAS.2008.4541553

CIE. (2004). *Colorimetry* (3. ed. ed.). CIE Central Bureau.

CMC colour-difference formula. (1984). *Color Research and Application, 9*(4), 250.

Coren, S., Ward, L., & Enns, J. (2003). *Sensation and Perception* (sixth ed.). John Wiley & Sons.

Cui, Z., & Gong, G. (2018). The effect of machine learning regression algorithms and sample size on individualized behavioral prediction with functional connectivity features. *NeuroImage (Orlando, Fla.), 178,* 622-637. 10.1016/j.neuroimage.2018.06.001

Dubey, A. K., & Jain, V. (2019). Comparative Study of Convolution Neural Network's Relu and Leaky-Relu Activation Functions. *Applications of Computing, Automation and Wireless Systems in Electrical Engineering* (pp. 873-880). Springer Singapore. 10.1007/978-981-13-6772-4_76

Duntley, S. Q. (1942). The Optical Properties of Diffusing Materials. *Journal of the Optical Society of America (1930), 32*(2), 61. 10.1364/JOSA.32.000061

Fairchild, M. D. (2013). *Color appearance models* (3. ed. ed.). Wiley.

Fearnley-Sander, D. (1979). Hermann Grassmann and the Creation of Linear Algebra. *The American Mathematical Monthly, 86*(10), 809.

Furferi, R., & Carfagni, M. (2010). Prediction of the Color and of the Color Solidity of a Jigger-dyed Cellulose-based Fabric: A Cascade Neural Network Approach. *Textile Research Journal, 80*(16), 1682-1696.

Furferi, R., Governi, L., & Volpe, Y. (2016). Color matching of fabric blends: hybrid Kubelka-Munk + artificial neural network based method. *Journal of Electronic Imaging, 25*(6), 061402.

Garay, H., Eterradossi, O., & Benhassaine, A. (2004). Predicting changes in the color of powders: Does Melamed's model fit to real industrial powders? *Color Research & Application: Endorsed by Inter-Society Color Council, the Colour Group (Great Britain), Canadian Society for Color, Color Science Association of Japan, Dutch Society for the Study of Color, the Swedish Colour Centre Foundation, Colour Society of Australia, Centre Français De La Couleur, 29*(6), 413-419.

Garrett, D. A., & Peters, R. H. (1956). 11-Effect of Penetration on Reflectance of Dyed Textile Fibres. *Journal of the Textile Institute Transactions, 47*(3), T166-T178. 10.1080/19447027.1956.10750388

Giuseppe, B. (2017). *Machine Learning Algorithms*. Packt Publishing Ltd.

Grossberg, S. (1988). Nonlinear neural networks: Principles, mechanisms, and architectures. *Neural Networks, 1*(1), 17-61. https://doi.org/10.1016/0893-6080(88)90021-4

Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., Liu, T., Wang, X., Wang, G., Cai, J., & Chen, T. (2018). Recent advances in convolutional neural networks. *Pattern Recognition, 77*, 354-377. https://doi.org/10.1016/j.patcog.2017.10.013

Guan, S. (2018). Fabric defect delaminating detection based on visual saliency in HSV color space. *Journal of the Textile Institute (2004), 109*(12), 1560-1573.

Günther, F., & Fritsch, S. (2010). Neuralnet: training of neural networks. *The R Journal, 2*(1), 30. 10.32614/RJ-2010-006

H. Jang, A. Park, & K. Jung. (2008). Neural Network Implementation Using CUDA and OpenMP. Paper presented at the - *2008 Digital Image Computing: Techniques and Applications,* 155-161. 10.1109/DICTA.2008.82

Hajipour, A., & Shams-Nateri, A. (2019). Improve neural network-based color matching of inkjet textile printing by classification with competitive neural network. *Color Research and Application, 44*(1), 65-72.

Han, J., & Moraga, C. (2005). The influence of the sigmoid function parameters on the speed of backpropagation learning. *From Natural to Artificial Neural Computation* (pp. 195-201). Springer Berlin Heidelberg. 10.1007/3-540-59497-3_175

Harrington, P. d. B. (1993). Sigmoid transfer functions in backpropagation neural networks. *Analytical Chemistry, 65*(15), 2167-2168.

HECHT-NIELSEN, R. (1992). III.3 - Theory of the Backpropagation Neural Network**Based on “nonindent” by Robert Hecht-Nielsen, which appeared in Proceedings of the

International Joint Conference on Neural Networks 1, 593–611, June 1989. © 1989 IEEE. In H. Wechsler (Ed.), *Neural Networks for Perception* (pp. 65-93). Academic Press. https://doi.org/10.1016/B978-0-12-741252-8.50010-8

Heggie, D., Wardman, R. H., & Luo, M. R. (1996). A comparison of the colour differences computed using the CIE94, CMC(l:c) and BFD(l:c) formulae. *Journal of the Society of Dyers and Colourists, 112*(10), 264-269.

Hemingray, C., & Westland, S. (2016). A novel approach to using neural networks to predict the colour of fibre blends. *Coloration Technology, 132*(4), 297-303.

Hung, O. N., Chan, C. K., Kan, C. W., & Yuen, C. W. M. (2017). An analysis of some physical and chemical properties of CO.sub.2 laser-treated cotton-based fabrics. *Cellulose (London), 24*(1), 363.

Hung, O. N., Song, L. J., Chan, C. K., Kan, C. W., & Yuen, C. W. M. (2011). Using artificial neural network to predict colour properties of laser-treated 100% cotton fabric. *Fibers and Polymers, 12*(8), 1069-1076.

Hung, O., Chan, C., Kan, C., & Yuen, C. (2017). Effect of the CO2 laser treatment on properties of 100% cotton knitted fabrics. *Cellulose, 24*(4), 1915-1926.

Hurvich, L. M., & Jameson, D. (1957). An opponent-process theory of color vision. *Psychological Review, 64*(6p1), 384-404. 10.1037/h0041403

Hwang, J. P., Kim, S., & Park, C. K. (2015). Development of a color matching algorithm for digital transfer textile printing using an artificial neural network and multiple regression. *Textile Research Journal, 85*(10), 1076-1082.

Janos, S. (1996). *Colorimetry* (2. ed., corr. reprint ed.). CIE.

Jasper, W. J., & Kovacs, E. T. (1994). *Using Neural Networks and NIR Spectrophotometry to Identify Fibers*

Jasper, W. J., Kovacs, E. T., & Berkstresser, G. A. (1993). Using Neural Networks to Predict Dye Concentrations in Multiple-Dye Mixtures. *Textile Research Journal, 63*(9), 545-551.

Jawahar, M., Narasimhan Kannan, C. B., & Kondamudi Manobhai, M. (2015). Artificial neural networks for colour prediction in leather dyeing on the basis of a tristimulus system. *Coloration Technology, 131*(1), 48-57.

Jawahar, M., Venba, R., Jyothi, G., Kanth, S. V., Doss, M. J., & Chandra Babu, N. K. (2013). *Dry colour prediction of leather from its wet state*. Wiley. 10.1111/cote.12033

Jebara, T. (2003). *Machine Learning*. Springer.

Jin, X., Xu, C., Feng, J., Wei, Y., Xiong, J., & Yan, S. (2015). Deep Learning with S-shaped Rectified Linear Activation Units. https://arxiv.org/abs/1512.07030

Jun, X., Wang, J., Zhou, J., Meng, S., Pan, R., & Gao, W. (2021). Fabric defect detection based on a deep convolutional neural network using a two-stage strategy. *Textile Research Journal, 91*(1-2), 130-142. 10.1177/0040517520935984

Kan, C., Kan, C., Song, L., & Song, L. (2016). An Artificial Neural Network Model for Prediction of Colour Properties of Knitted Fabrics Induced by Laser Engraving. *Neural Processing Letters, 44*(3), 639-650.

Khataee, A. R., & Mirzajani, O. (2010). UV/peroxydisulfate oxidation of C. I. Basic Blue 3: Modeling of key factors by artificial neural network. *Desalination, 251*(1-3), 64-69.

Kim, H. J., Youn, S., Choi, J., Kim, H., Shim, M., & Yun, C. (2021). Indexing surface smoothness and fiber softness by sound frequency analysis for textile clustering and classification. *Textile Research Journal, 91*(1-2), 200-218. 10.1177/0040517520935211

Kiryanov, I. I., Tulyabaev, A. R., Mukminov, F. K., & Khalilov, L. M. (2018). Neural network for prediction of 13C NMR chemical shifts of fullerene C60 mono-adducts. *Journal of Chemometrics, 32*(9), n/a.

Kubelka, P., & Munk, F. (1931). *An Article on Optics of Paint Layers*

Kulappurath, S. K. (2018). *Investigating the Role of Texture on Visual and Instrumental Color Difference Assessments.* http://www.lib.ncsu.edu/resolver/1840.20/35434

Kuo, C., & Fang, C. (2006). Optimization of the processing conditions and prediction of the quality for dyeing nylon and lycra blended fabrics. *Fibers and Polymers, 7*(4), 344-351.

Lathrop, A. L. (1965). Diffuse Scattered Radiation Theories of Duntley and of Kubelka–Munk. *Journal of the Optical Society of America (1930), 55*(9), 1097. 10.1364/JOSA.55.001097

Li, C., Liu, T., & Ye, W. (2020). A novel feature-based network with sequential information for textile defect detection. *Coloration Technology, 136*(6), 476-484. 10.1111/cote.12493

Li, H., Lai, L., Chen, L., Lu, C., & Cai, Q. (2015). The Prediction in Computer Color Matching of Dentistry Based on GA+BP Neural Network. *Computational and Mathematical Methods in Medicine, 2015*, 816719-7.

Li, P., Wang, J., & Jing, J. (2015a). Application of improved back propagation algorithm in color difference detection of fabric. *Color Research and Application, 40*(3), 311-317.

Li, P., Wang, J., & Jing, J. (2015b). Application of improved back propagation algorithm in color difference detection of fabric. *Color Research & Application, 40*(3), 311-317.

Li, S., Shamey, R., & Xu, C. (2009). Prediction of depth of shade of a dyed polyester fabric based on fibre fineness and fabric structure. *Coloration Technology, 125*(5), 296-303. 10.1111/j.1478-4408.2009.00210.x

Lin, J. (2013). *Factors Affecting the Perception and Measurement of Optically Brightened White Textiles* http://www.riss.kr/pdu/ddodLink.do?id=T13584393

Lopes, N., & Ribeiro, B. (2009). GPU Implementation of the Multiple Back-Propagation Algorithm. *Intelligent Data Engineering and Automated Learning - IDEAL 2009* (pp. 449-456). Springer Berlin Heidelberg. 10.1007/978-3-642-04394-9_55

Lorente-Leyva, L. L., Alemany, M. M. E., Peluffo-Ordóñez, D. H., & Araujo, R. A. (2021). Demand Forecasting for Textile Products Using Statistical Analysis and Machine Learning

Algorithms. *Intelligent Information and Database Systems* (pp. 181-194). Springer International Publishing. 10.1007/978-3-030-73280-6_15

Luo, J., Lu, K., Zhong, Y., Zhang, B., & Lv, H. (2021). Cashmere and wool identification based on convolutional neural network. *Journal of Engineered Fibers and Fabrics, 16*, 155892502110050. 10.1177/15589250211005088

Luo, M. R., & Rigg, B. (1986). Uniform Colour Space Based on the CMC(l:c) Colour-difference Formula. *Journal of the Society of Dyers and Colourists, 102*(5-6), 164-171.

Luo, M. R., Cui, G., & Rigg, B. (2001). The development of the CIE 2000 colour-difference formula: CIEDE2000. *Color Research and Application, 26*(5), 340-350.

M. M. Lau, & K. Hann Lim. (2018). Review of Adaptive Activation Function in Deep Neural Network. Paper presented at the - *2018 IEEE-EMBS Conference on Biomedical Engineering and Sciences (IECBES)*, 686-690. 10.1109/IECBES.2018.8626714

Mangine, H., Jakes, K., & Noel, C. (2005). A preliminary comparison of CIE color differences to textile color acceptability using average observers. *Color Research and Application, 30*(4), 288-294.

Marcus, R. T. (1997). *Colour Physics for Industry, second edition*. Wiley Subscription Services, Inc., A Wiley Company. 10.1002/(SICI)1520-6378(199712)22:63.0.CO;2-Y

Mark D. Fairchild. (2013). *Color Appearance Models* (3rd ed.). John Wiley & Sons, Ltd.

McDonald, R., & Smith, K. J. (2008). CIE94-a new colour-difference formula. *Journal of the Society of Dyers and Colourists, 111*(12), 376-379.

Melamed, N. T. (1963). Optical properties of powders. Part I. Optical absorption coefficients and the absolute value of the diffuse reflectance. Part II. Properties of luminescent powders. *Journal of Applied Physics, 34*(3), 560-570.

Melgosa, M., Cui, G., Oleari, C., Pardo, P. J., Huang, M., Li, C., & Luo, M. R. (2017). Revisiting the weighting function for lightness in the CIEDE2000 colour-difference formula. *Coloration Technology, 133*(4), 273-282.

Mudgett, P. S., & Richards, L. W. (1971). Multiple scattering calculations for technology. *Applied Optics (2004), 10*(7), 1485-1502. 10.1364/AO.10.001485

Mudgett, P. S., & Richards, L. W. (1972). Multiple scattering calculations for technology II. *Journal of Colloid and Interface Science, 39*(3), 551-567. 10.1016/0021-9797(72)90064-1

North, A. D., & Fairchild, M. D. (1993a). Measuring color-matching functions. Part I. *Color Research and Application, 18*(3), 155-162. 10.1002/col.5080180305

North, A. D., & Fairchild, M. D. (1993b). Measuring color-matching functions. Part II. New data for assessing observer metamerism. *Color Research and Application, 18*(3), 163-170. 10.1002/col.5080180306

Olimov, B., Karshiev, S., Jang, E., Din, S., Paul, A., & Kim, J. (2021). Weight initialization based-rectified linear unit activation function to improve the performance of a convolutional neural network model. *Concurrency and Computation, 33*(22), n/a. 10.1002/cpe.6143

Opponent-process theory of colour vision. (2009).

Oulton, D. P., & Westland, S. (2017). Vector-based modelling of colour difference: a pilot study of the DE2000 colour difference model. *Coloration Technology, 133*(1), 15-25.

PANTONE textile color guide (Paper ed. ed.)

PANTONE textile color specifier (Paper ed. ed.)

Paul Kubelka, & Franz Munk. (1931). Ein Beitrag zur Optik der Farbanstriche. *Zeits. F. Techn. Physik, 12*, 593-601.

Petrulis, D. (2014). Dyeing of Microfibres: Problems in Dye Demand Computations. *Fibres & Textiles in Eastern Europe,* (1 (103)), 115-118.

Pineo, O. W. (1940). Residual Photometric Errors in the Commercial Recording Spectrophotometer. *Journal of the Optical Society of America (1930), 30*(7), 276. 10.1364/JOSA.30.000276

Pribbenow, J., Mejauschek, M., Landgraf, P., Grund, T., Bräuer, G., & Lampke, T. (2019). Neural network for prediction of hardness profiles for steel alloys after plasma nitriding. *IOP Conference Series. Materials Science and Engineering, 480,* 12019.

Quinlan, J. R. (1986). Induction of decision trees. *Machine Learning, 1*, 81-106. https://doi.org/10.1007/BF00116251

Radovan, A., & Ban, Z. (May 2018). Prediction of HSV color model parameter values of cloud movement picture based on artificial neural networks. Paper presented at the 1110-1114.

Randall, D. (1998). Instruments for the measurement of color. *Textile Chemist and Colorist, 30*, 20-26.

Reddy, M., Jasper, W. J., McGregor, R., & Lee, G. (1997). Effects of Temperature and Salt on Dye Mixtures in the Batch Dyeing Process. *Textile Research Journal, 67*(2), 109-117.

Riedmiller, M. (1994). Advanced supervised learning in multi-layer perceptrons — From backpropagation to adaptive learning algorithms. *Computer Standards and Interfaces, 16*(3), 265-278. 10.1016/0920-5489(94)90017-5

Russell, I. (2012). *The Delta Rule*

S. Fujii, & H. Hayashi. (2019). Comparison of Performance by Activation Functions on Deep Image Prior. Paper presented at the - *2019 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*, 255-258. 10.1109/ICAIIC.2019.8669063

Saeed, U., Alsadi, J., Ahmad, S., Rizvi, G., & Ross, D. (2014). Polymer Color Properties: Neural Network Modeling. *Advances in Polymer Technology, 33*(S1), n/a.

Sayed, M., & Baker, F. (2015). E-Learning optimization using supervised artificial neural-network. *Journal of Software Engineering and Applications, 8*(01), 26.

SÈVE, R. (1996). Practical Formula for the Computation of CIE 1976 Hue Difference. *Color Research and Application, 21*(4), 314. 10.1002/col.5080210405

Sève, R. (1991). New formula for the computation of CIE 1976 Hue difference. *Color Research and Application, 16*(3), 217-218. 10.1002/col.5080160311

Shamey, R., & Zhao, X. (2014). *Modelling, Simulation and Control of the Dyeing Process*. Elsevier Science.

Shams-Nateri, A., Amirshahi, S. H., & Latifi, M. (2006). Prediction of Yarn Cross-Sectional Color from Longitudinal Color by Neural Network. *Research Journal of Textile and Apparel, 10*(2), 25-35.

Sharma, G., & Rodrguez-Pardo, C. E. (Jan 23, 2012). The dark side of CIELAB. Paper presented at the , *8292*(1) 82920D-10.

Sharma, G., Wu, W., & Dalal, E. N. (2005). The CIEDE2000 color-difference formula: Implementation notes, supplementary test data, and mathematical observations. *Color Research and Application, 30*(1), 21-30. 10.1002/col.20070

Shataee, S., Kalbi, S., Fallah, A., & Pelz, D. (2012). Forest attribute imputation using machine-learning methods and ASTER data: comparison of k-NN, SVR and random forest regression algorithms. *International Journal of Remote Sensing, 33*(19), 6254-6280. 10.1080/01431161.2012.682661

Shen, J., Zhou, X., Ma, H., & Chen, W. (2017). Spectrophotometric prediction of pre-colored fiber blends with a hybrid model based on artificial neural network and Stearns–Noechel model. *Textile Research Journal, 87*(3), 296-304.

Speranskaya, N. I. (1959). Determination of Spectrum Color Coordinates for 27 Normal Observers. *Optics and Spectroscopy, 7*, 424.

Stiles, W. S., & Burch, J. M. (1959). N.P.L. Colour-matching Investigation: Final Report (1958). *Optica Acta, 6*(1), 1-26. 10.1080/713826267

Ugur, O. C., & Behcet, B. B. (2019). Dependence of colour difference formulae on regular changes of colour coordinates in CIELAB colour space. *Industria Textilă (Bucharest, Romania : 1994), 70*(3), 248-254.

Veit, D. (2021). 3 - Neural networks in textile engineering. *Advances in Modeling and Simulation in Textile Engineering* (pp. 39-98). Elsevier Ltd. 10.1016/B978-0-12-822977-4.00017-0

Wardman, R. H., Farooq, S., & Smith, K. J. (2012). Determination of dyers' perceived components of colour difference (depth, brightness and hue) between two similar colours from their spectral reflectance values. *Coloration Technology, 128*(3), 161-168.

Wyszecki, G. W., & Stiles, W. S. (1982). *Color science : concepts and methods, quantitative data and formulae*. Wiley.

Wythoff, B. J. (1993). Backpropagation neural networks: A tutorial. *Chemometrics and Intelligent Laboratory Systems, 18*(2), 115-155. https://doi.org/10.1016/0169-7439(93)80052-J

Xiao, Q., Wang, R., Zhang, S., Li, D., Sun, H., & Wang, L. (2020). Prediction of pilling of polyester–cotton blended woven fabric using artificial neural network models. *Journal of Engineered Fibers and Fabrics, 15*, 155892501990015.

Yang, F., Cerrai, D., & Anagnostou, E. N. (2021). The Effect of Lead-Time Weather Forecast Uncertainty on Outage Prediction Modeling. *Forecasting, 3*(3), 501-516. 10.3390/forecast3030031

Yang, F., Wanik, D. W., Cerrai, D., Bhuiyan, M. A. E., & Anagnostou, E. N. (2020). Quantifying Uncertainty in Machine Learning-Based Power Outage Prediction Model Training: A Tool for Sustainable Storm Restoration. *Sustainability (Basel, Switzerland), 12*(4), 1525. 10.3390/su12041525

Yang, F., Watson, P., Koukoula, M., & Anagnostou, E. N. (2020). Enhancing Weather-Related Power Outage Prediction by Event Severity Classification. *IEEE Access, 8*, 60029-60042. 10.1109/ACCESS.2020.2983159

Yves Chauvin, & David E. Rumelhart. (2013). *Backpropagation*. Taylor and Francis. 10.4324/9780203763247

Zhao, S., Yin, L., Zhang, J., Wang, J., & Zhong, R. (2020). Real-time fabric defect detection based on multi-scale convolutional neural network. *IET Collaborative Intelligent Manufacturing, 2*(4), 189-196. 10.1049/iet-cim.2020.0062

Zhao, X., Zhang, M., & Zhang, J. (2021). Ensemble learning-based CNN for textile fabric defects classification. *International Journal of Clothing Science and Technology*, *33*(4), 664-678. 10.1108/IJCST-12-2019-0188

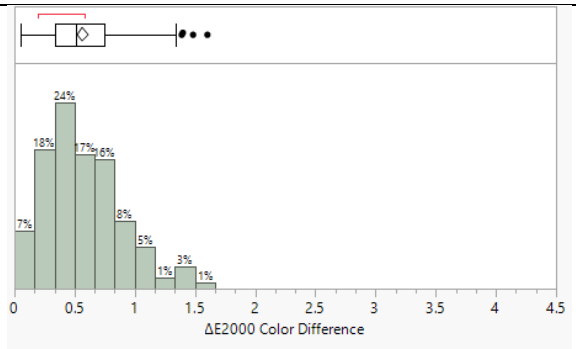
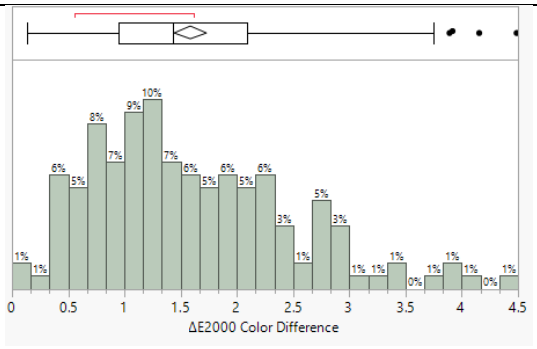
Zheng, D., Liu, Z., Zou, H., Xiong, Q., Liu, J., Wang, M., Liu, G., Pan, X., & Du, Z. (2021). Fuzzy clustering analysis of comprehensive hand of polyester fabric based on the CHES-FY system. *Textile Research Journal, 91*(7-8), 743-751. 10.1177/0040517520957409

Zou, J., Han, Y., & So, S. (2008). Overview of Artificial Neural Networks. *Artificial Neural Networks* (pp. 14-22). Humana Press. 10.1007/978-1-60327-101-1_2

Appendix

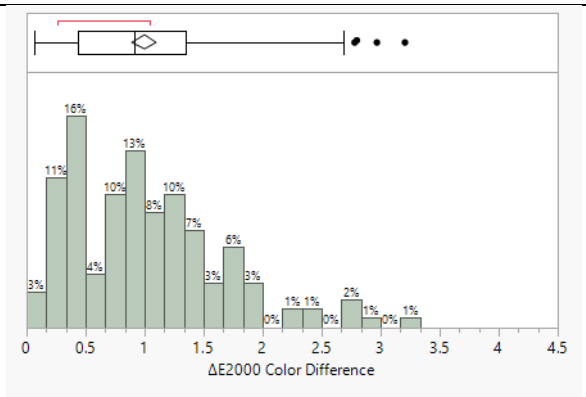
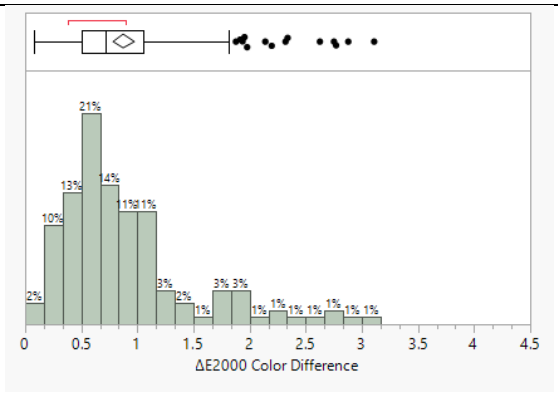
Appendix 1. Histogram of color difference for the models

Table A1(a). Color difference for models with 4 error functions

ΔE_{2000} Model		**ΔE_{1976} Model**	
			
Mean	**0.567**	**Mean**	**1.587**
Std Dev	**0.314**	**Std Dev**	**0.879**
Std Err Mean	**0.025**	**Std Err Mean**	**0.071**
Upper 95% Mean	**0.617**	**Upper 95% Mean**	**1.727**
Lower 95% Mean	**0.517**	**Lower 95% Mean**	**1.447**
N	**153**	**N**	**153**

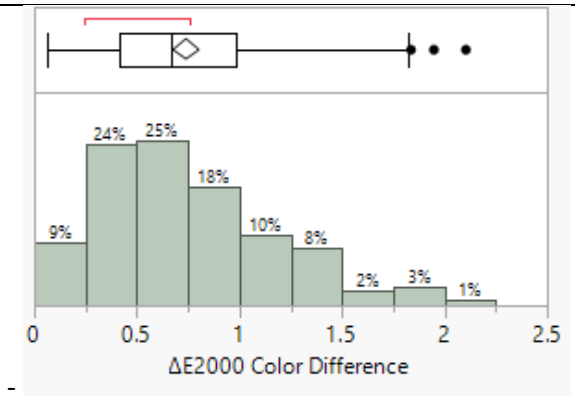
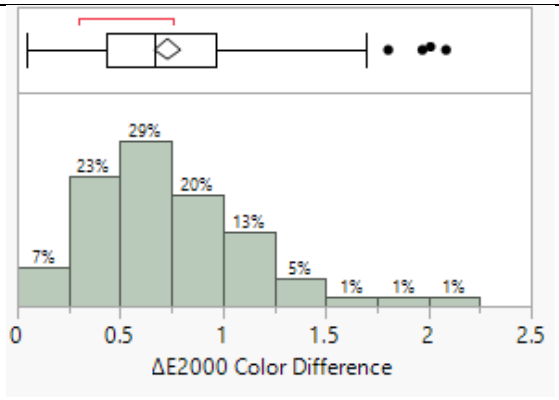
Note: The Figures above are the histograms and box plots for the CIE ΔE_{2000} color differences between the real and predicted testing color samples in dry state. The X-axis represents the value of CIE ΔE_{2000} color difference and the Y-axis represents the percentage of the testing samples has a ΔE_{2000} color difference value distributed in each interval.

Table A1(b). Color difference for models with 4 error functions

ΔE_{1994} Model		**ΔE_{CMC} Model**	
			
Mean	**0.997**	**Mean**	**0.879**
Std Dev	**0.642**	**Std Dev**	**0.594**
Std Err Mean	**0.052**	**Std Err Mean**	**0.048**
Upper 95% Mean	**1.099**	**Upper 95% Mean**	**0.974**
Lower 95% Mean	**0.894**	**Lower 95% Mean**	**0.784**
N	**153**	**N**	**153**

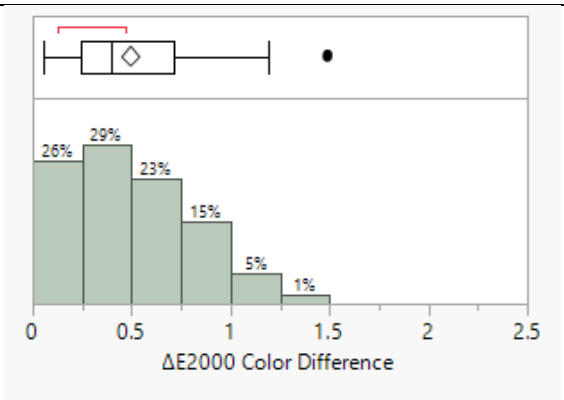
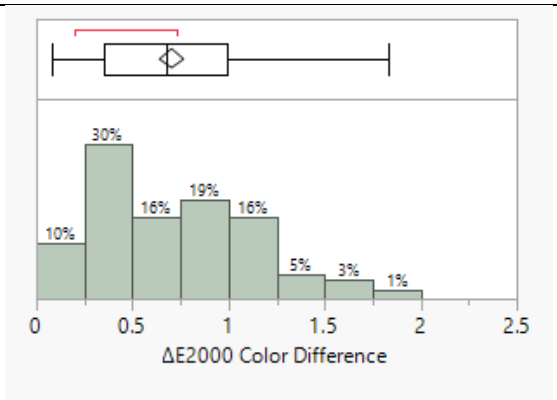
Note: The Figures above are the histograms and box plots for the CIE ΔE_{2000} color differences between the real and predicted testing color samples in dry state. The X-axis represents the value of CIE ΔE_{2000} color difference and the Y-axis represents the percentage of the testing samples has a ΔE_{2000} color difference value distributed in each interval.

Table A2(a). ΔE_{2000} statistics of 4 models under 4 bar pressure

3*6*3		3*7*3	
			
Mean	**0.739**	**Mean**	**0.729**
Std Dev	**0.417**	**Std Dev**	**0.392**
Std Err Mean	**0.034**	**Std Err Mean**	**0.032**
Upper 95% Mean	**0.806**	**Upper 95% Mean**	**0.792**
Lower 95% Mean	**0.673**	**Lower 95% Mean**	**0.666**
N	**153**	**N**	**153**

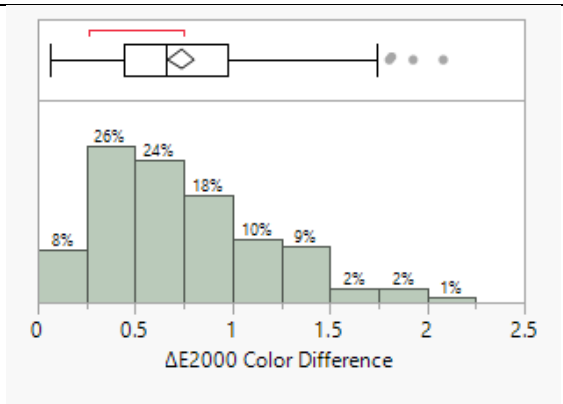
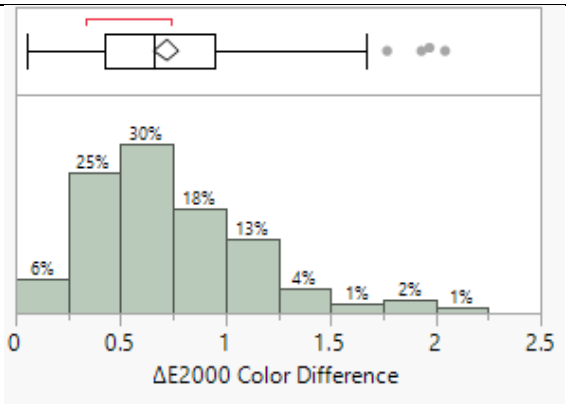
Note: The Figures above are the histograms and box plots for the CIE ΔE_{2000} color differences between the real and predicted testing color samples in dry state. The X-axis represents the value of CIE ΔE_{2000} color difference and the Y-axis represents the percentage of the testing samples has a ΔE_{2000} color difference value distributed in each interval.

Table A2(b). ΔE_{2000} statistics of 4 models under 4 bar pressure

3*8*3		3*9*3	
			
Mean	**0.496**	**Mean**	**0.705**
Std Dev	**0.304**	**Std Dev**	**0.396**
Std Err Mean	**0.025**	**Std Err Mean**	**0.032**
Upper 95% Mean	**0.545**	**Upper 95% Mean**	**0.768**
Lower 95% Mean	**0.448**	**Lower 95% Mean**	**0.642**
N	**153**	**N**	**153**

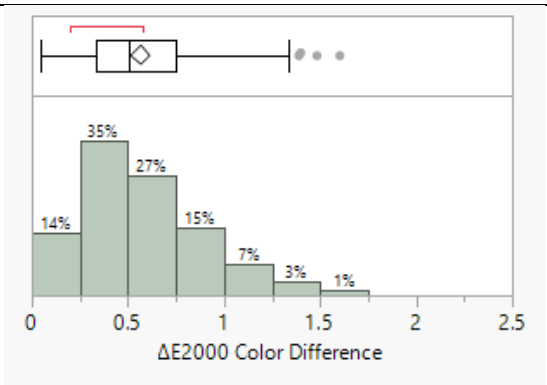
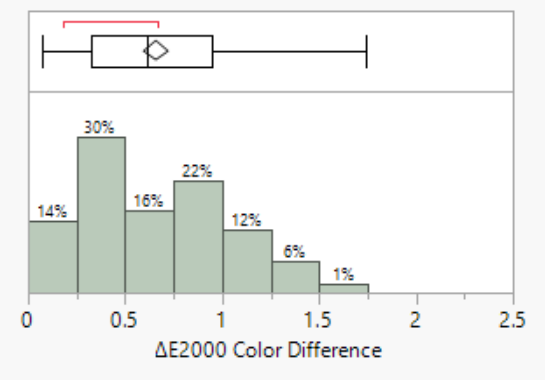
Note: The Figures above are the histograms and box plots for the CIE ΔE_{2000} color differences between the real and predicted testing color samples in dry state. The X-axis represents the value of CIE ΔE_{2000} color difference and the Y-axis represents the percentage of the testing samples that have a ΔE_{2000} color difference in that interval.

Table A3(a). ΔE2000 statistics of 4 models under 2 bar pressure

3*6*3		3*7*3	
			
Mean	**0.737**	**Mean**	**0.719**
Std Dev	**0.408**	**Std Dev**	**0.380**
Std Err Mean	**0.033**	**Std Err Mean**	**0.031**
Upper 95% Mean	**0.802**	**Upper 95% Mean**	**0.780**
Lower 95% Mean	**0.672**	**Lower 95% Mean**	**0.659**
N	**153**	**N**	**153**

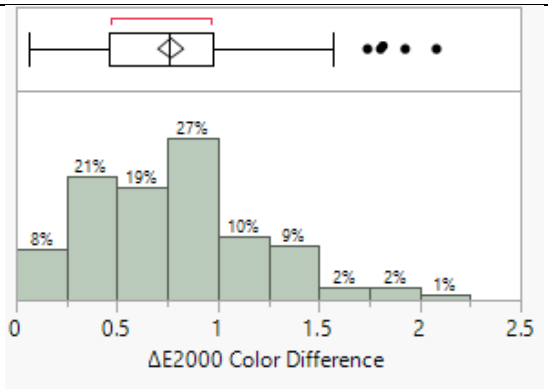
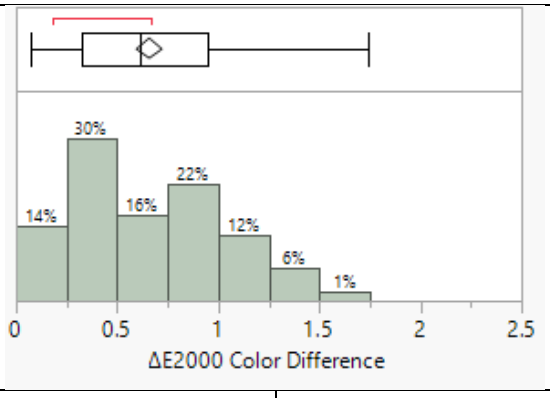
Note: The Figures above are the histograms and box plots for the CIE ΔE_{2000} color differences between the real and predicted testing color samples in dry state. The X-axis represents the value of CIE ΔE_{2000} color difference and the Y-axis represents the percentage of the testing samples has a ΔE_{2000} color difference value distributed in each interval.

Table A3(b). ΔE2000 statistics of 4 models under 2 bar pressure

3*8*3		3*9*3	
			
Mean	**0.567**	**Mean**	**0.657**
Std Dev	**0.314**	**Std Dev**	**0.381**
Std Err Mean	**0.025**	**Std Err Mean**	**0.031**
Upper 95% Mean	**0.617**	**Upper 95% Mean**	**0.718**
Lower 95% Mean	**0.517**	**Lower 95% Mean**	**0.596**
N	**153**	**N**	**153**

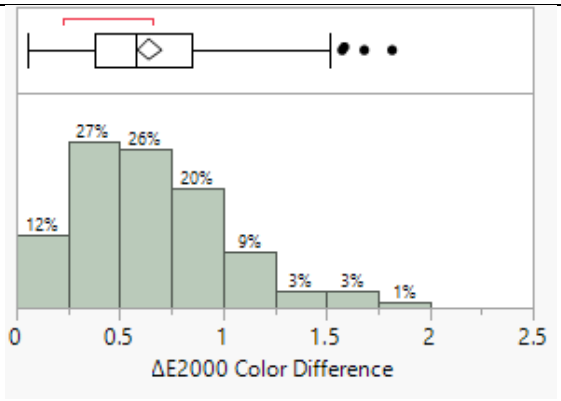
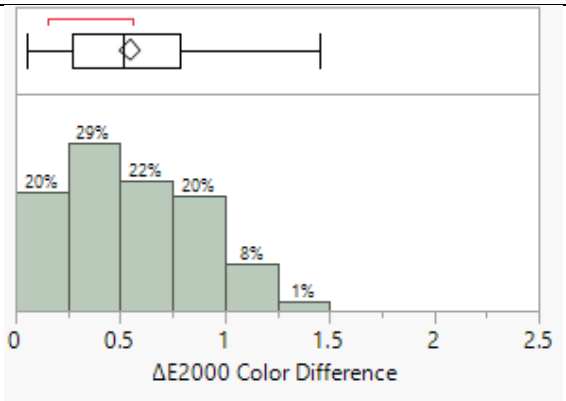
Note: The Figures above are the histograms and box plots for the CIE ΔE_{2000} color differences between the real and predicted testing color samples in dry state. The X-axis represents the value of CIE ΔE_{2000} color difference and the Y-axis represents the percentage of the testing samples has a ΔE_{2000} color difference value distributed in each interval.

Table A4(a). ΔE2000 statistics of 4 models under 1 bar pressure

3*6*3		3*7*3	
			
Mean	**0.767**	**Mean**	**0.657**
Std Dev	**0.402**	**Std Dev**	**0.381**
Std Err Mean	**0.032**	**Std Err Mean**	**0.031**
Upper 95% Mean	**0.831**	**Upper 95% Mean**	**0.718**
Lower 95% Mean	**0.703**	**Lower 95% Mean**	**0.596**
N	**153**	**N**	**153**

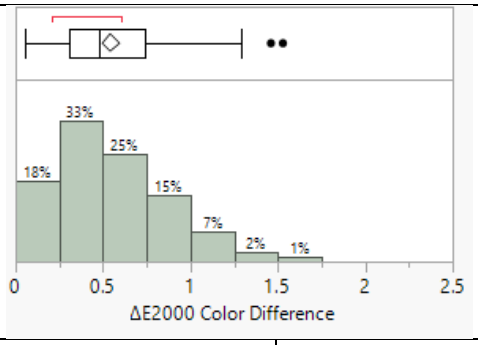
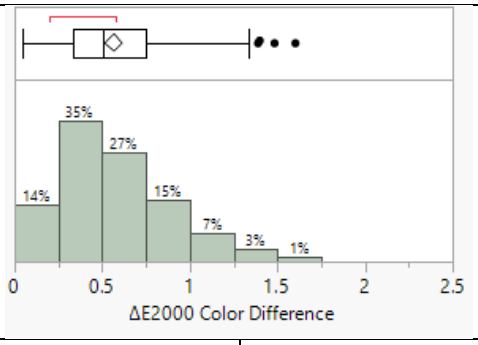
Note: The Figures above are the histograms and box plots for the CIE ΔE_{2000} color differences between the real and predicted testing color samples in dry state. The X-axis represents the value of CIE ΔE_{2000} color difference and the Y-axis represents the percentage of the testing samples has a ΔE_{2000} color difference value distributed in each interval.

Table A4(b). ΔE2000 statistics of 4 models under 1 bar pressure

3*8*3		3*9*3	
			
Mean	**0.643**	**Mean**	**0.547**
Std Dev	**0.356**	**Std Dev**	**0.318**
Std Err Mean	**0.029**	**Std Err Mean**	**0.026**
Upper 95% Mean	**0.700**	**Upper 95% Mean**	**0.598**
Lower 95% Mean	**0.586**	**Lower 95% Mean**	**0.497**
N	**153**	**N**	**153**

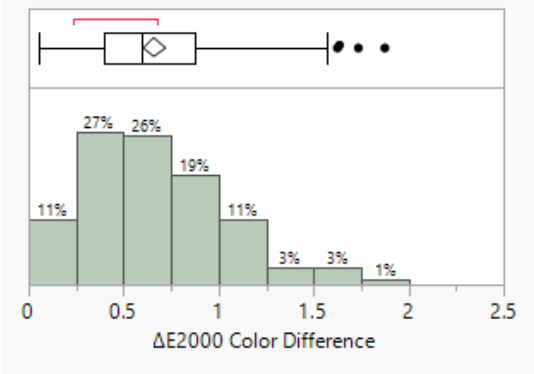
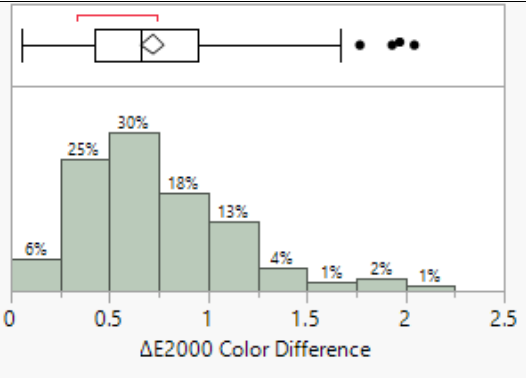
Note: The Figures above are the histograms and box plots for the CIE ΔE_{2000} color differences between the real and predicted testing color samples in dry state. The X-axis represents the value of CIE ΔE_{2000} color difference and the Y-axis represents the percentage of the testing samples has a ΔE_{2000} color difference value distributed in each interval.

Table A5(a). ΔE2000 statistics of 4 models under 0.5 bar pressure

3*6*3		3*7*3	
			
Mean	**0.542**	**Mean**	**0.567**
Std Dev	**0.311**	**Std Dev**	**0.314**
Std Err Mean	**0.025**	**Std Err Mean**	**0.025**
Upper 95% Mean	**0.592**	**Upper 95% Mean**	**0.617**
Lower 95% Mean	**0.492**	**Lower 95% Mean**	**0.517**
N	**153**	**N**	**153**

Note: The Figures above are the histograms and box plots for the CIE ΔE_{2000} color differences between the real and predicted testing color samples in dry state. The X-axis represents the value of CIE ΔE_{2000} color difference and the Y-axis represents the percentage of the testing samples has a ΔE_{2000} color difference value distributed in each interval.

Table A5(b). ΔE2000 statistics of 4 models under 0.5 bar pressure

3*8*3		3*9*3	
			
Mean	**0.663**	**Mean**	**0.719**
Std Dev	**0.367**	**Std Dev**	**0.380**
Std Err Mean	**0.029**	**Std Err Mean**	**0.030**
Upper 95% Mean	**0.721**	**Upper 95% Mean**	**0.780**
Lower 95% Mean	**0.604**	**Lower 95% Mean**	**0.659**
N	**153**	**N**	**153**

Note: The Figures above are the histograms and box plots for the CIE ΔE_{2000} color differences between the real and predicted testing color samples in dry state. The X-axis represents the value of CIE ΔE_{2000} color difference and the Y-axis represents the percentage of the testing samples has a ΔE_{2000} color difference value distributed in each interval.

Appendix 2. Example of HPC batch script

```
#!/bin/tcsh

## Identify the core number used in hpc. As suggested from NCSU IT, too many cores us
ed for one project may affect other hpc users. Therefore, I selected 24 cores for my
project. However, sometimes I was required to reduce the core number because of too m
any projects run at the same time.

#BSUB -n 24


## Identify the maximum running time (unit: minute) for one project. If the running t
ime reaches the maximum time, hpc will terminate the running code immediately.

#BSUB -W 1200


## BSUB -x  # use exclusive if memory intensive


## Name the output file and error file separately

#BSUB -o out.%J

#BSUB -e err.%J


## Give hpc commend to run R with the R script provided.

Module load R

## use 3*9*3 model for 2 bar pressure as an example

Rscript 9nudes2bar.R
```

Appendix 3. Example of R script for neural network

```
## we need use neuralnet package to build the models
library(neuralnet)


# set directory and read in the raw data
setwd("C:/Users/hanchi/Desktop/phd researcj/final round")
mydata<-read.csv("1bardata.csv",row.names = NULL)


#clean the data
data<-mydata[,c(1:8)]
colnames(data)<-c("Date_Number","L_dry","a_dry","b_dry","Pressure",'L_Wet','a_Wet','b
_Wet')


#delete all the null and empty rows
data<-data[!(is.na(data$b_wet) | data$b_wet==""), ]


#randomly shuffle the data, you can pick a different shuffle
set.seed(10)
data<-data[sample(nrow(data)),]


#output the raw data
data_train<-data[1:610,] # 90% of data
data_test<-data[610:762,]
write.table(data_test,'1test data.csv',sep = "\t",row.names=FALSE)
write.table(data_train,'1train data.csv',sep = "\t",row.names=FALSE)


# normalized function
normalize <- function(x) {
  return ((x - min(x)) / (max(x) - min(x)))
}
```

```
# Custom Error function with DE94 as an example
cie94 <- function(colvec1, colvec2) {
  # destructure color vectors into component L, a, b
  L1 <- colvec1[,1]
  a1 <- colvec1[,2]
  b1 <- colvec1[,3]
  L2 <- colvec2[,1]
  a2 <- colvec2[,2]
  b2 <- colvec2[,3]
  C1<-sqrt(a1^2 + b1^2)
  C2<-sqrt(a2^2 + b2^2)
  Da<-a1-a2
  Db<-b1-b2
  DL<-L1-L2
  DC<-C1-C2
  DH<-sqrt(Da^2 + Db^2 -DC^2)
  # for textiles K1=0.048,K2=0.014
  K1=0.048
  K2=0.014
  SL<-1
  SC<-1+K1*C1
  SH<-1+K2*C1
  DE94<-sqrt((DL/2)^2 + (DC/SC)^2 + (DH/SH)^2)
  return(DE94)
}


# normalized L* a* b* data

data[,c(6)]<-normalize(data[,c(6)])

data[,c(7)]<-normalize(data[,c(7)])

data[,c(8)]<-normalize(data[,c(8)])




#seperate the dataset into training group and testing group

data_train<-data[1:610,] # 80% of data

data_test<-data[610:762,]


# fit neural network, in this example, we create a 3*9*3 neural network model with st
epmax set at 1e+10 and threshold set at 0.01. And the error function is customed by u
sing DE2000 color difference functions.

pred<-neuralnet(data$L_dry + data$a_dry + data$b_dry ~ L_Wet +a_Wet+ b_Wet,

                data=data_train,hidden=9,stepmax=1e+10,threshold=0.01,

                err.fct='DE2000',linear.output=T)

save.image(file = "1pred.RData", version = NULL, safe = TRUE)
```

```
summary(pred)


# run the testing group to test the models

Predict<-predict(pred,data_test)

data_test["predict_dry"]<-Predict


# output the final results in L* a* b* value to calculate the predicted accuracy of t
he created model

write.table(data_test,'1bar.txt',sep = "\t",row.names=FALSE)
```