fibers

Article

Advanced Image Analysis and Machine Learning Models for Accurate Cover Factor and Porosity Prediction in Knitted Fabrics: Tailored Applications in Sportswear, Swimwear, and Casual Wear

Tomislav Rolich [1], Daniel Domović [1,*], Goran Čubrić [2] and Ivana Salopek Čubrić [3]

1   Department of Fundamental Natural and Engineering Sciences, University of Zagreb Faculty of Textile Technology, 10000 Zagreb, Croatia; tomislav.rolich@ttf.unizg.hr
2   Department of Clothing Technology, University of Zagreb Faculty of Textile Technology, 10000 Zagreb, Croatia; goran.cubric@ttf.unizg.hr
3   Department of Textile Design and Management, University of Zagreb Faculty of Textile Technology, 10000 Zagreb, Croatia; ivana.salopek@ttf.unizg.hr
*   Correspondence: daniel.domovic@ttf.hr

Abstract: This paper presents a study focused on developing robust algorithms for cover factor and porosity calculation through digital image analysis. Computational models based on machine learning for efficient cover factor prediction based on fabric parameters have also been developed. Five algorithms were devised and implemented in MATLAB: the single threshold algorithm (ST); multiple linear threshold algorithms, ML-1 and ML-2; and algorithms with multiple thresholds obtained by the Otzu method, MT-1 and MT-2. These algorithms were applied to knitted fabrics used for football, swimming, and leisure. Algorithms ML-1 and MT-1, employing multiple thresholds, outperformed the single threshold algorithm. The ML-1 variant yielded the highest average porosity value at 95.24%, indicating the importance of adaptable thresholding in image analysis. Comparative analysis revealed that algorithm variants ML-2 and MT-2 obtain lower cover factors compared to ML-1 and MT-1 but can detect potential void areas in fabrics with higher reliability. Algorithm MT-1 proved to be the most sensitive when it came to distinguishing between different fabric samples. Computational models that were developed based on random tree, random forest, and SMOreg machine learning algorithms predicted cover factor based on fabric parameters with up to 95% accuracy.

Keywords: knitted fabric; cover factor; porosity; digital image analysis; machine learning models; regression; MATLAB; Weka

1. Introduction

Definition of porosity. Participating in sports is a daily routine for many individuals, whether they are amateurs or professionals. The selection of suitable and comfortable sportswear is a crucial element in achieving peak performance. Comfort, defined as the psychological sensation experienced during physical activity in the current environmental conditions, is paramount. Knitted fabrics are preferred in sportswear manufacturing due to their comfortable properties, characterized by high stretchability and low stress. Given that sports activities lead to the generation of heat and sweat through the contact of clothing with the body, it becomes essential to dissipate heat and release water vapor into the environment, influencing the wearer's sense of comfort. This phenomenon is intricately tied to the porosity of the garment [1].

The porosity of a fabric is defined as the ratio of the total void area of the fabric to the total surface area of the fabric. Porosity responds to various parameters such as fabric thickness, pore shape and size, and the distribution of space between threads.

Importance of porosity in clothing comfort. Porosity plays a pivotal role in enhancing clothing comfort through its multifaceted impact on various key factors. Primarily, the relationship between porosity and air permeability is crucial, as higher porosity allows for improved airflow through fabrics. This contributes significantly to heat dissipation and the maintenance of a comfortable microclimate between the fabric and the wearer's skin.

Air permeability is an important factor in fabric comfort as it plays a role in the transport of moisture from the skin to the outside atmosphere, as water vapor travels mainly through the interstitial spaces of the fabric by diffusing air from one side of the fabric to the other [2]. Air permeability depends on the structural parameters of the fabric and porosity, as described by Havlova and Špánková [3].

Moreover, porosity influences moisture management by facilitating efficient wicking and evaporation of sweat, preventing the discomfort associated with damp or wet clothing. Benltoufa et al. confirmed that the amount of liquid absorbed by the garment, as well as the thermal comfort of clothing, are closely related to the pore size and distribution [4].

Beyond functional aspects, the flexibility and stretchability of knitted fabrics are often heightened with higher porosity, enhancing ease of movement. Additionally, the overall feel and texture of a fabric are influenced by the arrangement and distribution of pores, creating a subjective sense of lightness and breathability. Ultimately, porosity is instrumental in tailoring clothing to be adaptable to different activities, ensuring that individuals experience optimal comfort across diverse settings and physical exertions.

Digital image analysis in porosity study. To delve deeper into exploration of porosity, digital image analysis emerges as an invaluable tool, offering a nuanced perspective on fabric structure. In the field of digital image analysis porosity is often investigated in terms of a cover factor. If porosity represents the total void area of the fabric, cover factor is the opposite measure, representing the proportion of the fabric's surface covered by fibers or yarns.

Since the 1980s, it has been an ongoing mission of scientists to use image analysis for the evaluation of textile properties, including the porosity and description of pore size [5]. The first attempts used to obtain images for textile evaluation were based on the use of a scanner, a microscope with a CCD camera, or a digital camera with a lens. Further attempts were focused towards calculated optical porosity by analyzing white and black images [6,7].

In the most recent studies, Ramatan et al. used digital image analysis to measure pore size. The study also included the investigation of the effect of pore size, pore volume, and air permeability on fluid retention and air transfer [1].

The comparison of different algorithms that can be used to calculate the approximate porosity value of the fabric, such as air permeability calculation, digital image analysis, and geometric modeling, was a main topic of scientific interest of Bentoulfa et al. Their investigation indicated that the air permeability method is best for stretched fabric structures, digital image analysis for fabrics with high porosity, and geometric modeling for determining the shape of any structure's conformation [4].

Fouda et al. presented a method of image analysis for the determination of pore size distribution [8]. Another group of scientists developed a theoretical model to predict the porosity and air permeability of the knitted fabric before production as a function of the following geometrical parameters: cross-sectional area of each pore, depth of each pore or fabric thickness, and number of pores per unit area [2]. It has been shown that the proposed model can be used to predict porosity with high reliability.

Researchers also developed a computer program to calculate porosity based on the threshold used to convert a digital image from a grayscale to a binary scale and calculate pore size by counting pixels [9].

When conducting the image analysis, image processing is extremely important to reduce the loss of data on the structure and size of individual pores. Owczarek described the influence and importance of image pre-treatment methods for interstitial pore (ITP) detection with the aim of accurately identifying each ITP tissue structure using an opti-

mal pre-treatment algorithm for further quantitative morphometric structural analysis of specialized fabrics [10]. As indicated, the main problem with ITP identification in image analysis was the misclassification of pixels that were close to the edges of the yarn, resulting in image noise among other problems.

Research focus. Against this backdrop, our research endeavors to contribute to this evolving dialogue on textile comfort by contributing to the digital image analysis field with application in knitted fabrics. We present five digital image analysis algorithms—the single threshold algorithm (ST) and two multiple threshold algorithms (ML and MT, each with two variants). The introduction of five new algorithms for calculating porosity is driven by the need for precision and applicability to the specific context of knitted fabric used for football, swimming, and outdoor wear. While existing algorithms may effectively calculate porosity in general scenarios, the intricacies of knitted fabrics necessitate tailored approaches.

Moreover, traditional methods for calculating cover factor and porosity of knitted fabrics can face difficulty when dealing with the inherent complexity of fabric images. This complexity includes variations in color, texture, and structure. To overcome this, we have developed multiple computational models, leveraging machine learning algorithms. These algorithms can investigate the intricacies of the samples and uncover patterns that may not be immediately apparent. Computational models developed in this research can predict the cover factor value based on the parameter specific for each knitted fabric. Through this approach, we aimed to improve the accuracy and efficiency of cover factor and porosity calculation in the textile industry.

2. Materials and Methods

2.1. Fabrics

A set of representative knitted fabrics was carefully selected for use in the manufacture of sportswear and casual clothing to be worn as a single layer against the skin. The fabrics were chosen to cover various applications and were categorized based on their primary purpose:

- Fabrics for football clothing: samples F1–F9, composed of 100% polyester, were selected for this series.
- Fabrics for swimwear: samples S1–S9, made of a blend of polyamide and elastane, were chosen.
- Fabrics for casual wear: samples I1–I4, consisting of various fiber compositions such as viscose, cotton, and modal blended with elastane, were included.

Table 1 provides an overview of the selected fabrics, along with their assigned fabric IDs and respective fiber compositions.

In the experimental part of this study, all fabrics were subjected to testing for mass per unit area and fabric thickness. The mass per unit area was determined by weighing a specimen sized 100×100 mm on an analytic scale. Fabric thickness was measured according to the ISO 5084:2003 standard [11], using flat parallel metal plates with a thickness gauge and applying a pressure of 1 kPa during the test. Ten measurements were performed at different locations on each sample to ensure representative data.

The measurements of mass per unit area and fabric thickness were carried out using the Dino-lite AM7915MZT Edge microscope (Dino-Lite, Almere, The Netherlands), which offers a resolution of 5 megapixels and a magnification range of 10–220 times. The average values of mass per unit area m and fabric thickness t for the selected fabrics are presented in Table 2, alongside their standard deviation SD. The obtained fabric images were subsequently utilized in the experimental part of the study.

Table 1. Overview of selected fabrics.

Nr.	Fabric ID	Fibre Composition	
		Main Yarn	Plating Yarn
1	F1	100% polyester	-
2	F2	100% polyester	-
3	F3	100% polyester	-
4	F4	100% polyester	-
5	F5	100% polyester	-
6	F6	100% polyester	-
7	F7	100% polyester	-
8	F8	100% polyester	-
9	F9	91% polyester	9% elastane
10	S1	80% polyamide	20% elastane
11	S2	80% polyamide	20% elastane
12	S3	78% polyamide	22% elastane
13	S4	78% polyamide	22% elastane
14	S5	59% polyamide	41% elastane
15	S6	73% polyamide	27% elastane
16	S7	80% polyamide	20% elastane
17	S8	72% polyamide	28% elastane
18	S9	71% polyamide	29% elastane
19	I1	93% viscose	7% elastane
20	I2	90% cotton	10% elastane
21	I3	89% modal	11% elastane
22	I4	91% cotton	9% elastane

Table 2. Mass per unit area and thickness of selected fabrics.
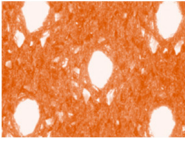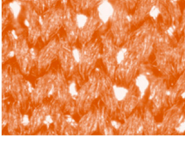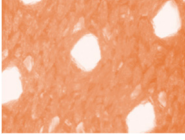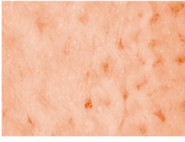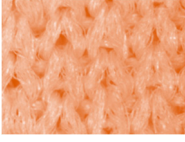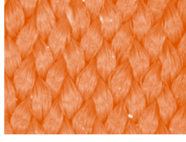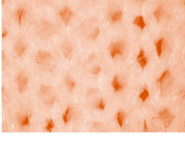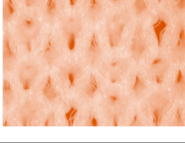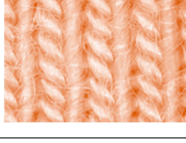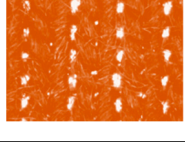
Nr.	Fabric ID	Image	m, g/m² SD	t, mm SD	Nr.	Fabric ID	Image	m, g/m² SD	t, mm SD
1	F1		100 0.03323	0.548 0.0063	12	S3		198 0.01134	0.561 0.0032
2	F2		145 0.00223	0.632 0.0092	13	S4		209 0.00127	0.527 0.0116
3	F3		178 0.05112	0.830 0.0047	14	S5		226 0.03555	0.495 0.0085

<p align="center">**Table 2.** *Cont.*</p>

Nr.	Fabric ID	Image	m, g/m² SD	t, mm SD	Nr.	Fabric ID	Image	m, g/m² SD	t, mm SD
4	F4		135 0.05775	0.739 0.0120	15	S6		113 0.00647	0.358 0.0063
5	F5		125 0.04469	0.488 0.0042	16	S7		216 0.01244	0.502 0.0103
6	F6		157 0.02754	0.618 0.0042	17	S8		160 0.03456	0.404 0.0097
7	F7		155 0.03099	0.491 0.0137	18	S9		184 0.03371	0.450 0.0170
8	F8		145 0.02618	0.556 0.0107	19	I1		223 0.06001	0.679 0.0320
9	F9		180 0.05941	0.492 0.0155	20	I2		135 0.11001	0.689 0.0080
10	S1		187 0.00768	0.655 0.0053	21	I3		160 0.09002	0.672 0.0131
11	S2		171 0.00713	0.655 0.0108	22	I4		170 0.05000	0.695 0.0170

We selected these fabrics to represent the diverse range of fabrics commonly used in sportswear, swimwear, and casual wear. By including fabrics intended for different purposes, such as football clothing, swimwear, and casual wear, we aimed to capture the variations in fabric composition, structure, and performance that are relevant to these specific applications. This selection allows for a comprehensive evaluation of porosity characteristics and their impact on comfort across different fabric series. The use of a wide range of fabrics ensures that the research findings are applicable to various types of clothing and provides insights that can inform the design and manufacturing of optimized sportswear fabrics.

2.2. Methods

2.2.1. Measuring Porosity Using Cover Factor

Porosity, in the context of knitted fabrics, refers to the extent of void spaces or gaps within a fabric structure. It is defined as the ratio of the total void area to the total surface area of the fabric:

$$P = \mathrm{area_{void}}/\mathrm{area_{total}} \tag{1}$$

Porosity is an important factor in determining the comfort and functionality of textiles, influencing characteristics such as breathability, moisture management, and thermal comfort.

Cover factor is a measure used in textile engineering to quantify the amount of space occupied by the fibers in a fabric. It represents the ratio of the area covered by fibers to the total surface area of the fabric. A higher cover factor indicates that more of the fabric's surface is covered by fibers, implying less open space or voids.

The relationship between cover factor (CF) and porosity (P) is mathematically expressed as:

$$\mathrm{CF} = 1 - \mathrm{P} \tag{2}$$

Fabrics with high porosity often have low cover factors, meaning there is more openness or void space between fibers or yarns. Fabrics with low porosity have higher cover factors, indicating that a larger proportion of the fabric is covered by fibers. Both CF and P are expressed as percentage.

The results in this study are presented in terms of cover factor, a common metric used in textile engineering and fabric analysis. Cover factor is a straightforward measure that directly reflects the coverage of fibers on the fabric surface. It is used to provide a clear and easily interpretable representation of the fabric structure.

2.2.2. Algorithms

This section outlines the five developed algorithms for calculating the porosity factor: the simple threshold algorithm (ST) and two variations of multiple thresholds algorithm, the multiple thresholds algorithm (ML) and the multiple thresholds algorithm with Otsu (MT). Programs were implemented in MATLAB [12].

All the developed algorithms employ a series of steps to convert a microscope photograph into a grayscale representation and subsequently calculate the porosity factor, presented in Figure 1. Each step serves a specific purpose and contributes to the overall accuracy and reliability of the algorithm:

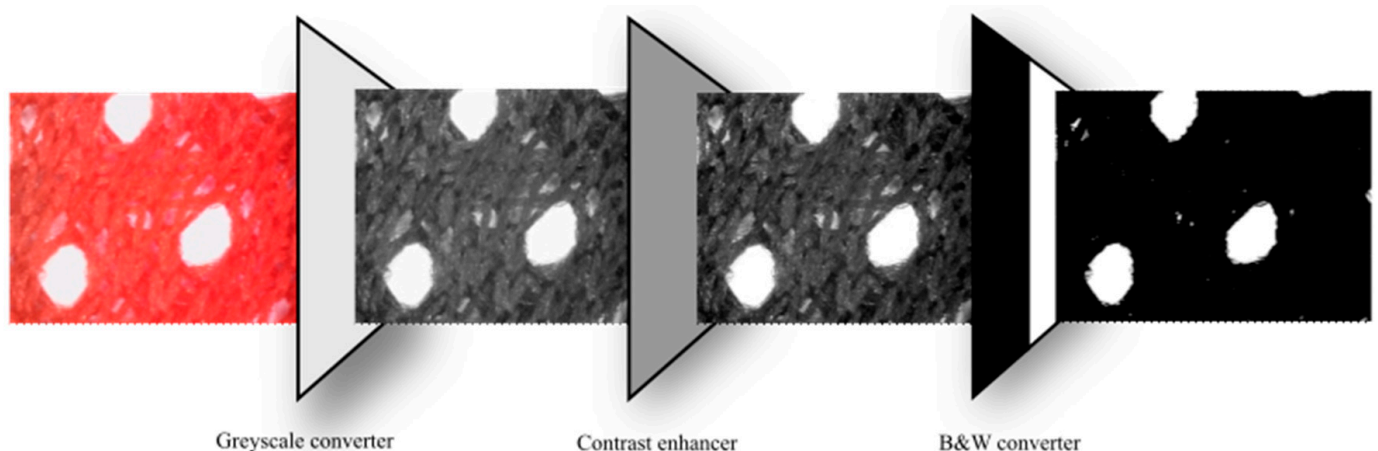Greyscale converter   Contrast enhancer   B&W converter

Figure 1. The steps for converting a photograph taken with a microscope to a black and white photograph include utilizing a grayscale converter, enhancing contrast, and converting to black and white using thresholding.

1. Grayscale Conversion:

The conversion of the RGB color space to grayscale is necessary to simplify the subsequent porosity calculations. By transforming the image into grayscale, algorithms eliminate color variations that may not directly correlate with the presence of porosity. The weighted average expression used for grayscale conversion follows NTSC formula:

$$GP = 0.2989R + 0.5870G + 0.1140B \tag{3}$$

with GP being the gray scale value of a pixel, and R, G, and B being the corresponding red, green, and blue values of a pixel in the original image [13].

2. Contrast Enhancement:

Enhancing the contrast in the grayscale photograph is crucial for accurate porosity analysis. By adjusting the gray levels to span the entire grayscale range (0–255 or 0–1), the algorithm maximizes the distinction between different gray values. This adjustment helps to reveal subtle variations in intensity, which may correspond to different degrees of porosity in the sample. A wider range of gray levels provides a more detailed representation of the image, allowing for more precise porosity calculations.

3. Thresholding:

The conversion from grayscale to a binary black-and-white image through thresholding is a fundamental step in the simple threshold algorithm. It categorizes pixels as black or white based on whether their grayscale values fall below or equal to a threshold value. The thresholding operation simplifies the image by creating a clear differentiation between the porous and non-porous regions, which is essential for subsequent pixel counting and cover factor calculation.

4. Pixel Counting and Cover Factor Calculation:

Once the binary image is obtained, the number of black pixels (representing non-porous regions) and white pixels (representing porous regions) are counted. The cover factor is then calculated as the percentage of white pixels relative to the total number of pixels in the image.

This calculation provides a quantitative measure of the porosity of the sample.

By following these steps, the developed algorithms effectively convert a microscope photograph into a grayscale representation, enhance contrast, isolate porous regions through thresholding, and quantify the porosity factor.

2.2.3. Simple Threshold Algorithm

Following the earlier described procedure, the simple threshold algorithm (ST) aims to convert a microscope photograph into a grayscale representation, which is essential for subsequent porosity calculations. The RGB color space is transformed into grayscale using a weighted average expression (1), whose (R, G, B) weights values of (0.2989, 0.5870, 0.1140) are derived from the standard formula for converting color images to grayscale.

To enhance the contrast in the grayscale photograph, the gray levels are adjusted to span the entire range from 0 to 255. Subsequently, the grayscale image is further converted into a binary black-and-white image using a fixed threshold value T = 200. The value of the hyper parameter T = 200 has been determined after thorough empirical analysis described in Section 4.4.

This thresholding operation categorizes pixels as black or white based on whether their grayscale values are lower or equal to T. The number of black pixels (B) and white pixels (W) are then counted in the resulting binary image. Finally, the cover factor (CF) is calculated as a percentage using Equation (1), which considers the ratio of white pixels to the total number of pixels.

However, we have recognized the potential limitations of a fixed threshold and addressed them by developing algorithms that employ an adaptive thresholding technique to

further expand on the ST algorithm. In this context, the simple threshold algorithm with T = 200 serves as a baseline for comparative purposes. By comparing the performance of the adaptive thresholding version (using Otsu's method) to the fixed threshold version, the effectiveness of the adaptive approach can be assessed and it can be determined whether superior segmentation results are provided.

2.2.4. Multiple Threshold Algorithms

To address the issues arising when using the single threshold algorithm, we have experimented with algorithms using multiple threshold values. We developed the ML (Multiple Threshold) and MT (Multiple Threshold with Otsu) algorithms as alternative approaches to calculate porosity. These algorithms offer additional flexibility and potential improvements over the simple threshold algorithm (ST) by introducing multiple thresholds and more refined pseudo-grayscale transformations.

The purpose of the ML and MT algorithms is to provide a more nuanced analysis of porosity by incorporating multiple thresholds instead of a single threshold value. This allows for a more detailed characterization of porosity within the sample, capturing variations in intensity that may correspond to different levels of porosity. The ML and MT algorithms aim to enhance the accuracy and precision of porosity calculations by introducing more sophisticated image processing techniques.

In the ML algorithm, the grayscale image is first linearly divided into eight equal parts to obtain multiple thresholds. This division allows for a more detailed differentiation of grayscale values. Each pixel is assigned a pseudo-grayscale value between 1 and 8 based on its corresponding threshold range T1 to T8.

To convert the pseudo-grayscale image into a binary black-and-white image, pixels with values from 1 to 7 are set to black, while pixels with a value of 8 are set to white. The number of black pixels (B) and white pixels (W) are then counted, and the cover factor is calculated using Equation (1).

A second version of the ML algorithm, ML-2, was created to further refine the porosity calculation. In this variation, pixels with pseudo-grayscale values from 1 to 6 are categorized as black, while those with a value of 7 or 8 are set to white to further experiment with enhancing segmentation accuracy. The remaining steps, including pixel counting and cover factor calculation, remain the same as in the previous version. Pixels with values of 7 and 8 represent the porosity in the knit, where pixels with a value of 8 indicate complete absence of porosity and those with a value of 7 denote potentially porous areas.

To further address the limitations that arise from using a fixed threshold set of ML algorithms, we have incorporated an adaptive thresholding technique that dynamically adjusts the threshold based on the image's characteristics, hence creating the MT algorithm. An adaptive thresholding technique, such as Otsu's, can automatically determine the optimal threshold value for each image, considering its specific features and reducing the reliance on a fixed threshold. This adaptability can enhance the algorithm's robustness and segmentation accuracy across a wider range of images and conditions.

The MT algorithm employs multiple thresholds to convert the grayscale image into a pseudo-grayscale image. Instead of linearly dividing the grayscale range, the Otsu method is utilized to dynamically determine the thresholds. The Otsu method, described in [14] and implemented in MATLAB, calculates optimal threshold values (T1 to T8) specific to each photograph by maximizing the between-class variance of pixel intensities, effectively separating the foreground and background regions. By applying Otsu's method, the algorithm can dynamically determine a threshold value that is better-suited to the image characteristics, improving the segmentation accuracy compared to using a fixed threshold.

The subsequent steps of assigning pseudo-grayscale values, converting to a binary black-and-white image, pixel counting, and cover factor calculation follow the same principles as in the ML algorithm. A second version of the MT algorithm was developed, in which the photograph is converted to black-and-white, so that all pseudo-grayscale values

with a value of 1 to 6 become black, and all pseudo-grayscale values with a value of 7 to 8 become white.

Figure 2 depicts the procedural steps and histograms of the grayscale and pseudo-grayscale images for the ML and MT algorithms and the differences on the original image each step produces. These visual representations aid in understanding the transformations and highlight the differences between the algorithms.
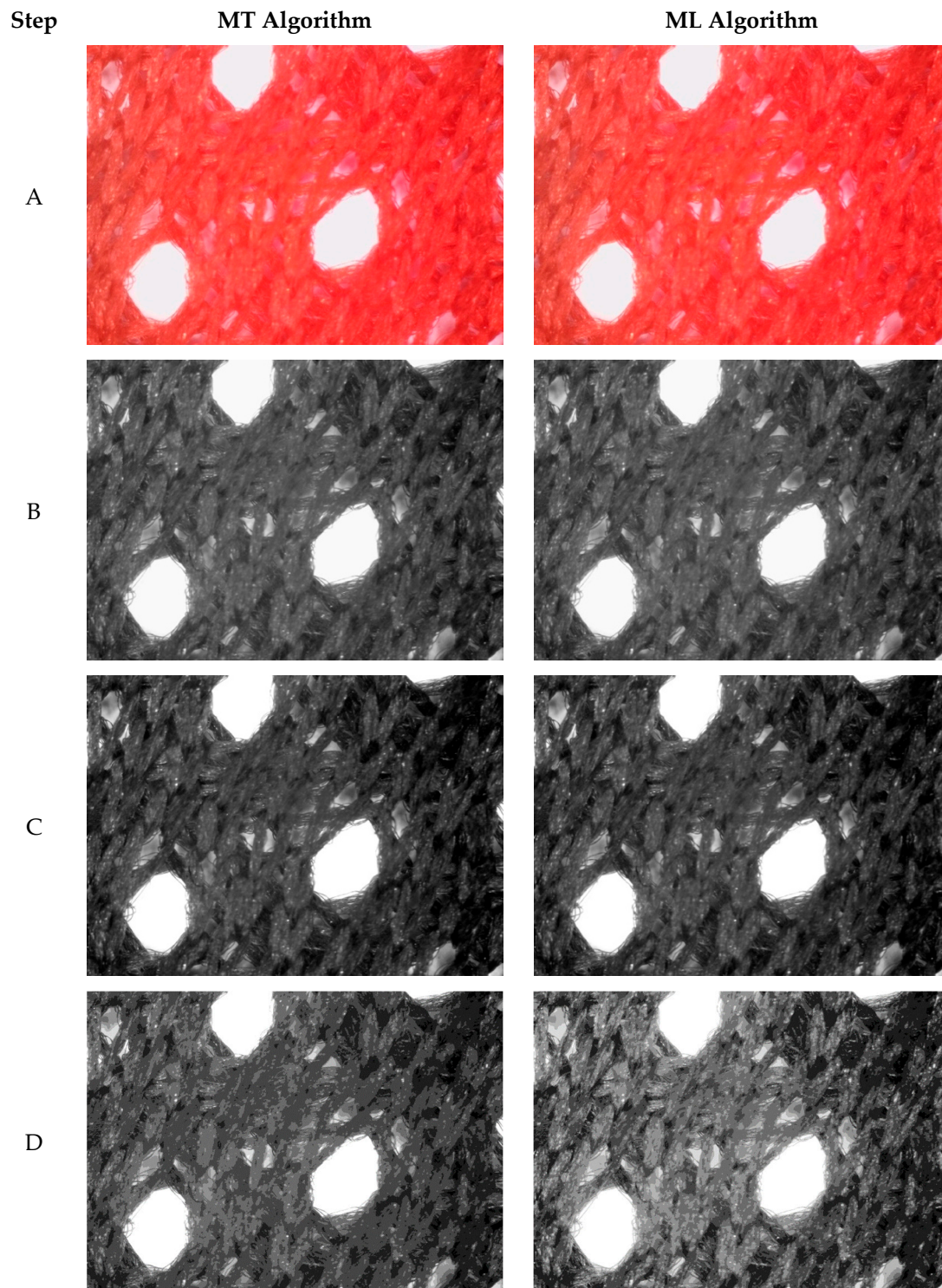
Figure 2. *Cont.*
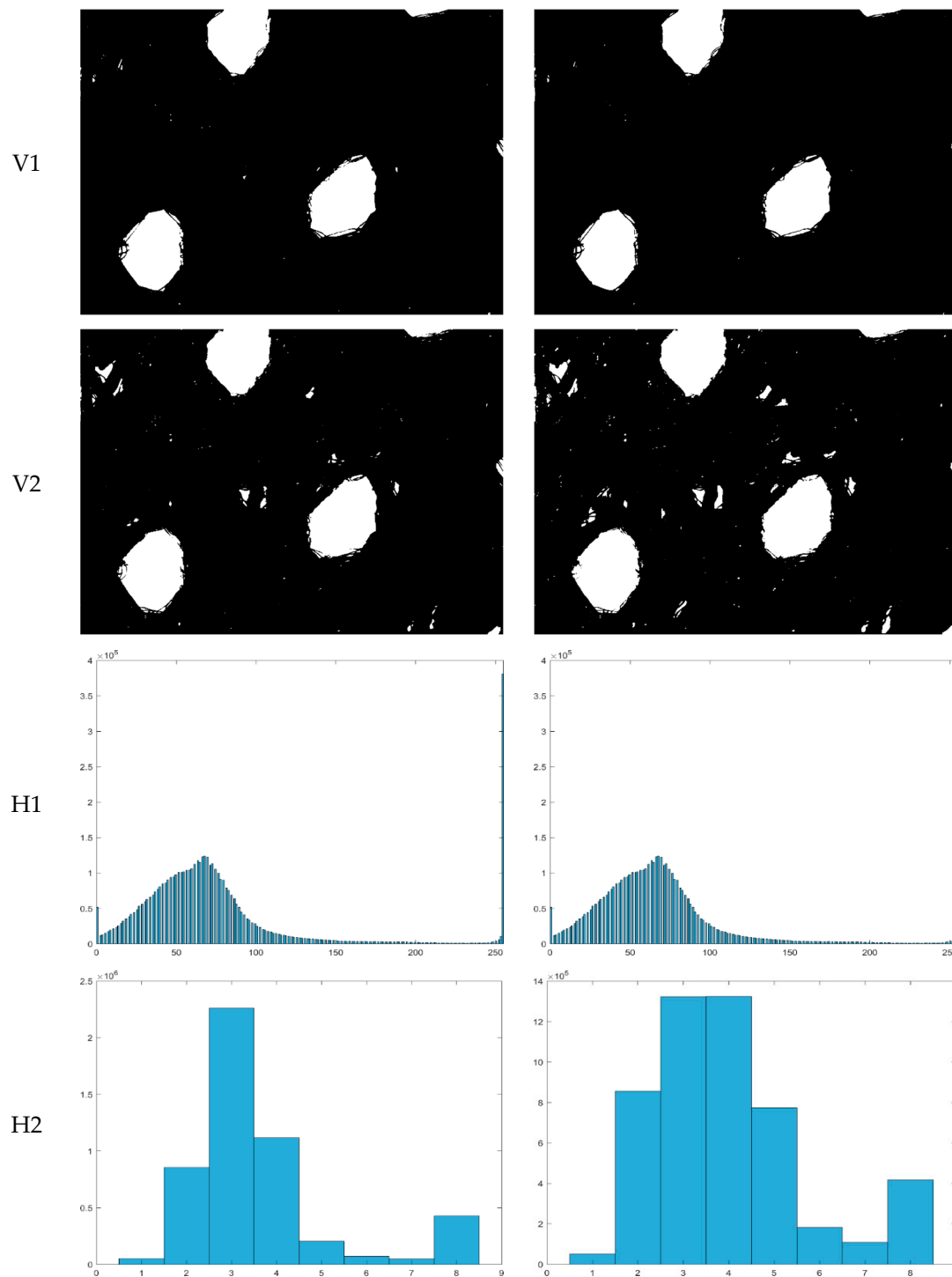
Figure 2. The figure depicts a digital image as it passes through the different stages of the ML and MT algorithms. The figure presents the original photo (**A**), grayscale photos (**B**), grayscale photos with enhanced contrast (**C**), pseudo-grayscale photos (**D**), black-and-white photos obtained by ML-1 and MT-1 algorithms (**V1**), and by ML-2 and MT-2 algorithm (**V2**), grayscale photos histogram (**H1**), and pseudo-grayscale photo histogram (**H2**).

The key difference between the ML and MT algorithms lies in the determination of threshold values and the subsequent pseudo-grayscale transformations. In the ML algorithm, the grayscale range is divided into eight equal parts, and threshold values (T1 to T8) are linearly distributed. These thresholds are empirically chosen to create a

pseudo-grayscale image with values ranging from 1 to 8, while in the MT algorithm the grayscale range is divided into eight parts using the Otsu method. The Otsu method automatically calculates threshold values (T1 to T8) based on the image's statistical properties, optimizing the separation of porous and non-porous regions.

The pseudocode for all algorithms is provided in Figure 3. It encompasses the procedures for obtaining the microscope photograph, enhancing image contrast, converting the photo to grayscale, and determining the cover factor using the appropriate formula based on the algorithm.

OBTAIN a photograph using a microscope

ENHANCE image contrast

CONVERT the photograph from RGB into grayscale using the expression (3)

IF algorithm = ST

    SET threshold to T = 200

    IF pixel's grayscale value < T

        SET pixel = black

    ELSE

        SET pixel = white

ELSE IF algorithm = ML1 **OR** algorithm = ML2

      OR algorithm = MT1 **OR** algorithm = MT2

      OBTAIN PSEUDO IMAGE

CALCULATE porosity using expression (1)

Figure 3. Pseudocode of ST, ML1, ML2, MT1, and MT2 algorithm.

Moreover, Figure 4 presents the pseudocode for the "OBTAIN PSEUDO IMAGE" function, an essential component of the ML and MT algorithms. This function calculates the thresholds and assigns pseudo-grayscale values according to the specific requirements of each algorithm.

FUNCTION _OBTAIN PSEUDO IMAGE_

IF algorithm = ML1 **OR** algorithm = ML2

    DISTRIBUTE grayscale range [0, 255] linearly into 8 parts to obtain 8 thresholds [T1, T8]

IF algorithm = MT1 **OR** algorithm = MT2

DISTRIBUTE grayscale range [0, 255] into 8 parts using the Otsu algorithm to obtain thresholds

[T1, T8]

DETERMINE pseudo-grayscale pixel value:

IF pixel's grayscale value <= T1 **THEN** pseudo-grayscale pixel value = 1 **ELSE**

IF pixel's grayscale value <= T2 **THEN** pseudo-grayscale pixel value = 2 **ELSE**

…

IF pixel's grayscale value <= T8 **THEN** pseudo-grayscale pixel value = 8

CONVERT obtained pseudo-grayscale image into black-and-white image:

IF algorithm = ML1 **OR** algorithm = MT1

IF (pseudo-grayscale pixel value = [1, 7]) pseudo_image_pixel_value = black

    ELSE pseudo_image_pixel_value = white

IF algorithm = ML2 **OR** algorithm = MT2

    IF (pseudo-grayscale pixel value = [1, 6]) pseudo_image_pixel_value = black

    ELSE pseudo_image_pixel_value = white

Figure 4. Pseudocode of _OBTAIN PSEUDO IMAGE_ function.

2.3. Machine Learning

In recent years, machine learning has emerged as a powerful tool in the field of computer science. It involves the development of data-driven models that can automatically

identify and recognize patterns from large sets of data. This unique ability has made machine learning an indispensable tool in fields such as image recognition, natural language processing, and financial forecasting [15].

One area where machine learning has shown great potential is in the analysis of digital images and for prediction models. The complexities and unique structures present in knitted fabrics can make it challenging for traditional computer algorithms to accurately analyze and classify them. However, with the adaptability and advanced pattern recognition abilities of machine learning models, these challenges can be overcome [16,17].

Through the utilization of machine learning algorithms, we have produced computational models that possess the ability to accurately predict the porosity values of knitted fabrics, without the necessity for complex digital image analysis. To achieve this, we utilized fabric parameters such as the mass per unit and thickness of the fabric as input values for the models, and the cover factor of the fabric as the output.

To create these models, we utilized the WEKA suite developed by the machine learning group at the University of Waikato, New Zealand. This suite offers a vast collection of algorithms and tools that enable researchers to build highly accurate and efficient predictive models [18].

For this study, we focused on regression models as they are best suited for numeric results and exclusively deal with numeric attributes. Some of the regression models used in this study include trees, such as alternating random tree and random forest, which utilize a decision tree algorithm to make predictions based on input data. Functions like linear regression, multilayer perceptron, and support vector machine (SMOReg) were also utilized to construct predictive models. In addition to these algorithms, we also employed lazy methods such as K-nearest neighbor classifier (Ibk) and K* (KStar), which use nearest neighbor matching to make predictions. Lastly, we incorporated meta techniques like additive regression, which combines multiple models to achieve a more accurate prediction.

2.4. Research Hypothesis

Based on the proposed algorithms and obtained material, the following research hypotheses have been posed:

H1. *Algorithms ML-1 and MT-1 that use adaptive thresholding will obtain higher cover factor than the basic algorithm with a fixed threshold ST-1.*

H2. *Algorithms with higher threshold segmentation, i.e., ML-2 and MT-2, will recognize the underlying knitted fabric structure and will obtain lower cover factor than their counterparts ML-1 and MT-1.*

H3. *MT-1 algorithm, which uses Otzu adaptive thresholding, will be the most sensitive algorithm to different fabric structures.*

H4. *A machine learning model can predict cover factor based on knitted fabric parameters with high accuracy.*

3. Results

3.1. Cover Factor Results per Sample in a Dataset

Multiple samples of knitted fabrics in the F, S, and I series were analyzed using five algorithms: ML-1, ML-2, MT-1, MT-1, and ST. The results, specifically cover factors, are summarized in Table 3, Table 4, and Table 5, respectively. The cover factor was calculated for a total of 22 knitted samples, each identified by a specific code (F1–F9, S1–S9, I1–I4). Multiple pictures of each sample were taken. Samples F1–F9 consisted of five photos each, as did samples S1, S3, S6, S8, and I1–I3. Samples S2, S4, S5, and S9 consisted of eight photos. Sample S7 consisted of ten photos and sample I4 consisted of four photos.

Table 3. Cover factors for samples in the F series.

Sample	ML-1 [%]	ML-2 [%]	MT-1 [%]	MT-2 [%]	ST [%]
		Algorithm for Cover Factor Calculation			
F1-1	86.30	83.13	86.63	82.04	84.77
F1-2	85.68	83.05	85.94	82.15	84.39
F1-3	84.79	82.67	85.45	83.03	83.76
F1-4	86.29	83.37	86.58	82.33	84.85
F1-5	85.10	82.73	85.52	82.55	83.93
F2-1	92.33	89.20	92.40	87.21	90.93
F2-2	94.01	91.40	93.90	88.65	92.86
F2-3	94.72	92.44	94.58	89.52	93.71
F2-4	97.92	95.96	97.59	92.89	97.02
F2-5	94.89	92.54	94.79	89.67	93.86
F3-1	91.51	90.56	91.71	89.55	91.09
F3-2	88.20	86.62	88.64	86.97	87.49
F3-3	88.96	87.66	89.30	87.37	88.35
F3-4	90.43	89.27	90.35	87.80	89.85
F3-5	88.04	86.57	88.10	85.50	87.37
F4-1	98.04	96.18	98.04	95.21	97.08
F4-2	98.22	96.63	97.99	94.11	97.61
F4-3	98.04	94.93	97.88	92.27	96.68
F4-4	98.17	96.57	98.31	95.46	97.58
F4-5	98.31	96.60	98.15	93.96	97.61
F5-1	97.27	93.79	97.06	91.40	95.84
F5-2	96.92	93.15	96.15	90.06	95.22
F5-3	97.54	93.96	97.24	91.40	95.90
F5-4	97.66	94.31	96.96	90.86	96.32
F5-5	97.32	93.72	96.57	89.96	95.69
F6-1	98.31	97.54	98.73	97.75	97.89
F6-2	98.09	96.86	97.77	94.56	97.45
F6-3	98.27	97.20	97.78	94.72	97.69
F6-4	98.27	96.89	98.42	96.79	97.67
F6-5	98.17	97.10	98.03	96.09	97.63
F7-1	97.94	95.61	97.60	93.37	96.86
F7-2	97.74	94.85	97.43	92.86	96.44
F7-3	97.95	95.96	97.83	94.54	97.02
F7-4	97.70	95.00	98.01	94.38	96.30
F7-5	97.93	95.75	98.04	95.06	97.03
F8-1	98.51	97.51	98.27	95.58	98.09
F8-2	98.40	97.61	98.86	97.82	97.99
F8-3	98.44	97.63	98.11	94.98	98.08

Table 3. *Cont.*

	Algorithm for Cover Factor Calculation				
Sample	ML-1 [%]	ML-2 [%]	MT-1 [%]	MT-2 [%]	ST [%]
F8-4	98.14	96.89	97.59	94.92	97.45
F8-5	98.34	97.43	98.18	95.60	97.91
F9-1	98.36	97.10	98.14	95.14	97.78
F9-2	98.48	97.05	98.70	95.80	97.90
F9-3	97.56	92.15	96.94	88.81	95.67
F9-4	98.01	94.38	97.39	89.45	96.82
F9-5	97.99	94.42	97.79	92.92	96.76

Table 4. Cover factors for samples in the S series.

	Algorithm for Cover Factor Calculation				
Sample	ML-1 [%]	ML-2 [%]	MT-1 [%]	MT-2 [%]	ST [%]
S1-1	98.36	97.14	98.41	96.86	97.80
S1-2	98.24	95.28	98.24	93.57	97.05
S1-3	98.04	95.07	97.62	92.31	96.69
S1-4	98.06	95.40	97.80	92.21	96.95
S1-5	98.19	96.83	98.19	95.37	97.62
S2-1	94.09	85.30	93.42	84.13	89.71
S2-2	94.28	86.89	93.75	84.86	90.52
S2-3	93.98	85.84	92.90	84.26	89.62
S2-4	93.54	85.17	93.22	84.47	88.98
S2-5	92.66	85.40	91.94	83.63	88.81
S2-6	93.15	84.63	93.15	84.63	88.48
S2-7	92.35	83.26	92.35	82.42	87.33
S2-8	92.82	85.19	91.66	83.74	88.72
S3-1	96.40	92.13	96.07	90.39	94.30
S3-2	96.33	92.28	96.00	90.70	94.28
S3-3	96.38	92.29	96.05	91.00	94.40
S3-4	96.47	92.52	96.15	90.89	94.52
S3-5	96.63	92.73	96.78	91.75	94.66
S4-1	95.54	85.00	94.70	80.88	91.44
S4-2	96.35	87.78	95.20	81.96	92.84
S4-3	95.75	84.20	94.18	77.19	90.93
S4-4	95.00	83.39	93.05	78.68	90.50
S4-5	95.77	82.14	94.05	76.98	90.22
S4-6	96.21	81.54	94.49	79.14	90.67
S4-7	95.60	82.73	94.01	77.49	90.41
S4-8	95.86	83.24	94.07	78.88	90.35
S5-1	86.86	56.04	83.29	57.36	72.00
S5-2	88.13	58.48	88.13	64.95	73.86

Table 4. *Cont.*

Sample	Algorithm for Cover Factor Calculation				
	ML-1 [%]	ML-2 [%]	MT-1 [%]	MT-2 [%]	ST [%]
S5-3	83.06	50.54	84.95	60.59	66.19
S5-4	83.53	54.27	81.53	56.96	68.81
S5-5	86.04	52.01	81.31	53.20	68.22
S5-6	88.05	56.39	84.55	57.80	73.18
S5-7	87.74	56.10	85.23	58.77	73.35
S5-8	87.57	53.04	85.79	55.58	70.44
S6-1	96.36	92.75	96.59	91.05	94.63
S6-2	96.04	91.28	95.16	88.47	93.79
S6-3	96.37	92.54	95.67	89.93	94.58
S6-4	96.54	92.68	96.02	90.08	94.72
S6-5	95.81	90.98	95.19	88.30	93.39
S7-1	78.30	61.32	78.30	61.32	69.14
S7-2	94.50	84.08	92.55	79.42	89.63
S7-3	78.13	60.57	79.67	60.57	68.71
S7-4	80.68	61.63	82.19	63.26	71.36
S7-5	78.05	58.52	79.67	60.12	68.21
S7-6	80.93	64.52	80.93	64.52	72.21
S7-7	94.27	84.97	92.64	80.64	89.75
S7-8	94.23	83.75	91.78	78.16	89.33
S7-9	93.61	83.03	92.33	79.88	88.81
S7-10	94.34	82.96	93.13	81.34	89.09
S8-1	98.14	96.40	98.68	96.72	97.27
S8-2	97.98	96.16	98.25	95.83	97.05
S8-3	98.07	96.66	98.36	96.06	97.36
S8-4	97.83	96.01	98.48	96.30	96.85
S8-5	97.95	96.08	98.35	96.25	97.04
S9-1	95.83	87.07	94.23	78.48	92.71
S9-2	95.31	86.10	93.51	78.86	91.87
S9-3	95.28	87.16	93.99	80.27	92.09
S9-4	95.78	88.21	94.72	83.04	92.75
S9-5	95.50	87.32	94.28	80.11	92.39
S9-6	94.16	78.01	92.30	73.04	88.04
S9-7	95.29	85.57	93.91	79.71	91.70
S9-8	95.61	88.67	94.10	83.37	92.54

Table 5. Cover factors for samples in the I series.

Sample	Algorithm for Cover Factor Calculation				
	ML-1 [%]	ML-2 [%]	MT-1 [%]	MT-2 [%]	ST [%]
I1-1	98.84	98.60	98.91	98.65	98.73
I1-2	98.75	98.39	98.99	98.75	98.57
I1-3	98.77	98.46	98.94	98.62	98.62
I1-4	98.70	98.19	98.77	98.19	98.43
I1-5	98.76	98.43	98.90	98.68	98.60
I4-1	96.58	95.91	96.96	96.50	96.24
I4-2	97.95	97.50	98.17	97.92	97.72
I4-3	94.76	93.96	94.61	92.65	94.36
I4-4	98.53	97.94	98.98	98.46	98.22
I4-5	97.87	97.24	98.50	97.84	97.55
I2-1	98.23	96.63	98.14	95.41	97.51
I2-2	98.05	95.77	97.53	92.85	97.02
I2-3	98.19	96.08	98.19	95.32	97.08
I2-4	98.22	96.70	98.51	96.37	97.48
I2-5	97.99	95.96	98.20	95.16	97.11
I3-1	96.88	88.68	95.57	82.83	93.82
I3-2	97.19	89.14	96.09	83.18	94.53
I3-3	96.82	88.73	95.54	83.12	93.81
I3-4	97.06	92.56	96.44	87.34	95.52
I3-5	97.64	93.11	97.24	89.26	96.05

Table 3 features results for nine knitted samples in series F, labeled F1, F2, ..., F9. Each row in the table corresponds to a specific sample from knitted samples, identified as F1-1, F1-2, ..., F9-5, depending on how many photos of a knitted sample were obtained. The columns represent different algorithms for cover factor calculation. In each field a cover factor obtained for a sample using one of the five algorithms is noted. The value of a cover factor is expressed as a percentage. For example, for sample F1-1, the cover factor values obtained using the ML-1, ML-2, MT-1, MT-2, and ST algorithms are 86.30%, 83.13%, 86.63%, 82.04%, and 84.77%, respectively.

We can generalize that higher cover factors are preferable since they indicate lower porosity (although this generalization will be discussed and analyzed later in the paper).

Tables 4 and 5 consist of cover factors for the knitted samples in the S and I series, respectively. The structure of these tables is similar to Table 3, with rows representing different samples and columns representing different algorithms.

Additional insights are provided by averaging the cover factors for all photos in each of the samples. Table 6 presents the resulting cover factors obtained from all algorithms, accompanied by descriptive statistical parameters:

- Arithmetic mean (AVG): the average cover factor value for each algorithm and sample.
- Standard deviation (STDEV): the measure of the dispersion or variability of the cover factor values for each algorithm and sample.
- Relative standard deviation (REL STDEV): the standard deviation expressed as a percentage of the arithmetic mean.
- Minimum (MIN): the lowest cover factor value observed for each algorithm across all samples.

- Maximum (MAX): the highest cover factor value observed for each algorithm across all samples.
- Range (RANGE): the difference between the maximum and minimum values for each algorithm.

Table 6. Descriptive statistics for cover factor values for all samples.

Sample	ML-1 [%]	ML-2 [%]	MT-1 [%]	MT-2 [%]	ST [%]	AVG	STDEV	REL STDEV	MIN	MAX	RANGE
F1	85.63	82.99	86.02	82.42	84.34	84.28	1.60	1.89%	82.35	86.02	3.68
F2	94.77	92.31	94.65	89.59	93.68	93.00	2.16	2.32%	89.59	94.79	5.20
F3	89.43	88.14	89.62	87.44	88.83	88.69	0.94	1.06%	87.37	89.64	2.27
F4	98.16	96.18	98.08	94.20	97.31	96.79	1.66	1.72%	94.20	98.18	3.98
F5	97.34	93.79	96.79	90.74	95.80	94.89	2.69	2.84%	90.74	97.34	6.61
F6	98.22	97.12	98.14	95.98	97.67	97.43	0.98	1.00%	95.94	98.34	2.40
F7	97.85	95.43	97.78	94.04	96.73	96.37	1.65	1.71%	94.04	97.93	3.89
F8	98.36	97.41	98.20	95.78	97.90	97.53	1.09	1.12%	95.74	98.46	2.72
F9	98.08	95.02	97.79	92.42	96.99	96.06	2.38	2.49%	92.42	98.12	5.70
S1	98.18	95.94	98.05	94.06	97.22	96.69	1.73	1.79%	94.06	98.19	4.12
S2	93.36	85.21	92.80	84.02	89.02	88.88	4.27	4.81%	84.02	93.36	9.34
S3	96.44	92.39	96.21	90.95	94.43	94.08	2.40	2.55%	90.95	96.47	5.53
S4	95.76	83.75	94.22	78.90	90.92	88.71	7.19	8.11%	78.90	95.76	16.86
S5	86.37	54.61	84.35	58.15	70.76	70.85	14.63	20.69%	54.61	86.61	32.00
S6	96.22	92.05	95.73	89.56	94.22	93.56	2.77	2.96%	89.56	96.27	6.71
S7	86.71	72.54	86.32	70.92	79.62	79.22	7.53	9.85%	70.60	87.17	16.57
S8	97.99	96.26	98.42	96.23	97.12	97.20	1.01	1.03%	96.07	98.42	2.35
S9	95.35	86.01	93.88	79.61	91.76	89.32	6.51	7.32%	79.61	95.35	15.73
I1	98.76	98.41	98.90	98.58	98.59	98.65	0.20	0.20%	98.41	98.90	0.49
I4	97.14	96.51	97.44	96.67	96.82	96.92	0.47	0.49%	96.25	97.47	1.23
I2	98.14	96.23	98.11	95.02	97.24	96.95	1.35	1.40%	95.02	98.24	3.21
I3	97.12	90.44	96.18	85.14	94.75	92.73	4.95	5.36%	85.14	97.12	11.98
MIN	85.63	54.61	84.35	58.15	70.76						
AVG	95.24	89.94	94.90	88.20	92.80						
MAX	98.76	98.41	98.90	98.58	98.59						

The table presents cover factor values (%) obtained by five algorithms (ML-1, ML-2, MT-1, MT-2, and ST) for different samples within three fabric series (F, S, and I).

Detailed analysis of parameters including the average, standard deviation, variance, minimum, maximum and range of each sample per each method is presented in Supplementary Table S1.

Based on the results displayed in Table 6, we observed that ML-1 generally has higher cover factor values compared to ML-2, MT-1, MT-2, and ST, across most samples. There is considerable variability in cover factor values within each fabric series and algorithm.

In the examination of fabric series, distinct trends emerge. In the football clothing (F series), ML-1 consistently exhibits elevated cover factor values, surpassing other algorithms. Notably, Sample F5 stands out with the highest range of cover factor values, reaching 6.61%. The swimwear (S series) also showcases ML-1's tendency towards higher cover factor values, with Sample S5 demonstrating the most considerable variability, boast-

ing a wide cover factor range of 32.00%. The differences are larger for samples S2, S4, S5, S7, S9, and I3. It should be noted that for these samples we encountered difficulties in determining cover factor. The fabric in sample S7 was dark. In samples S5 and S9, the fabric was two-colored, dark blue and white, which further complicated the calculation of porosity factor. Other challenges included the fact that the contrast algorithm greatly altered the photo (sample S4), the fabric was relatively porous (sample S2), or the structure of the fabric was irregular (sample I3), all of which made it very difficult to determine what was knitted and what was not (i.e., holes in the mesh). A more reliable cover factor could be calculated with the simple threshold algorithm, but with a higher threshold value than T = 200.

In contrast, the casual wear (I series) reveals a nuanced picture. While ML-1, ML-2, and ST exhibit comparable cover factor values, Sample I3 within this series displays a noteworthy range of cover factor values, indicating substantial variability at 11.98%. These fabric series analyses underscore the algorithmic nuances in capturing porosity, offering valuable insights into the distinctive characteristics of each series.

In algorithmic comparison, ML-1 consistently exhibits the highest average cover factor values, followed by ML-2, MT-1, MT-2, and ST. ML-2, MT-1, MT-2, and ST generally show lower variability (lower STDEV and REL STDEV) compared to ML-1.

To sum up, ML-1 tends to yield higher cover factor values compared to other algorithms. Variability in cover factor values suggests algorithmic sensitivity to the fabric structure or characteristics. Further statistical tests will be conducted to assess significant differences between algorithms. The observed differences may have implications for the choice of algorithm depending on the specific requirements of the application.

In Figure 5, box-and-whisker plots have been made, representing obtained data for five algorithms across all samples.
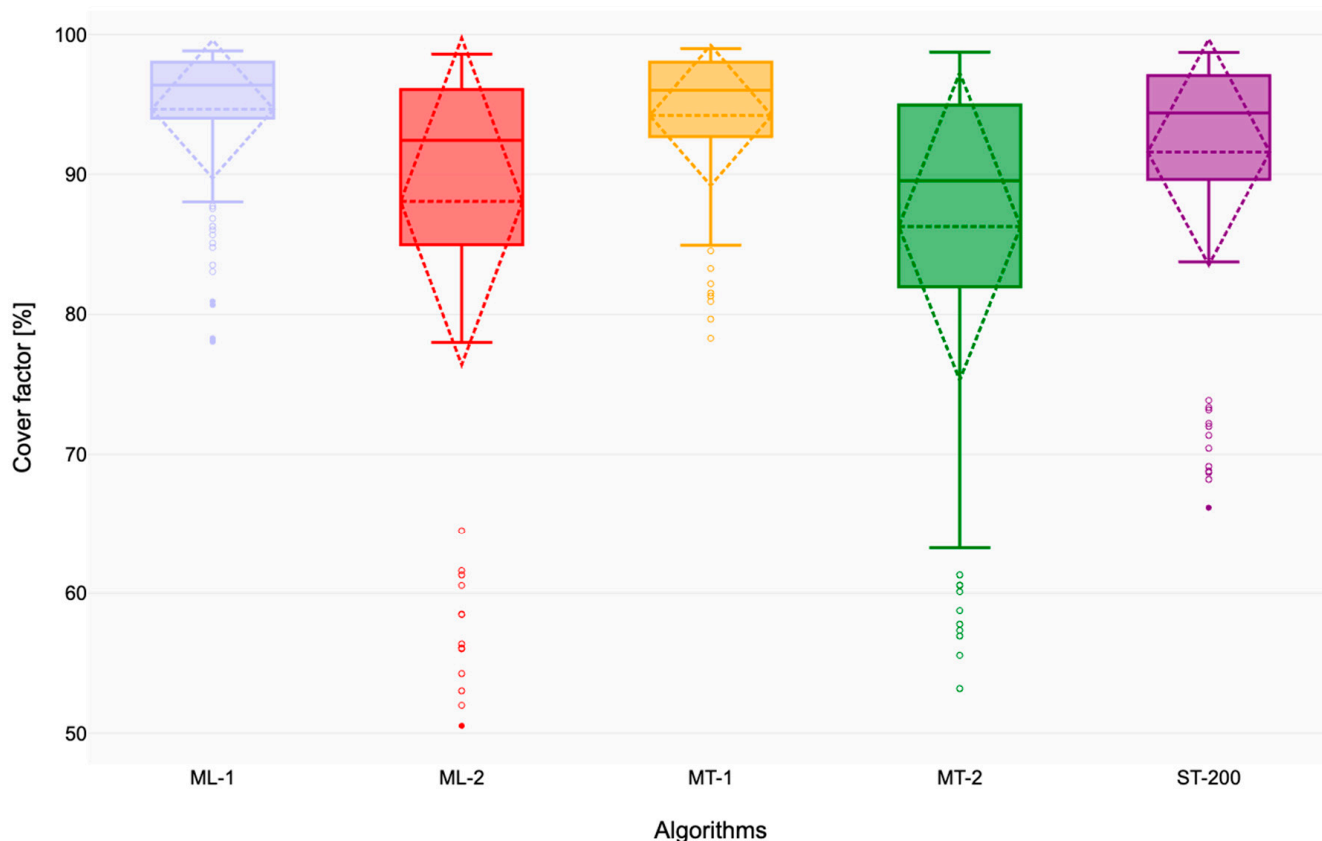

Figure 5. Box plot of ML-1, MT-1, and ST obtained through all samples.

3.2. Computational Models Based on Machine Learning for Cover Factor Prediction Based on Parameters of Knitted Fabric

In this section, we present computational models based on machine learning developed for predicting porosity in knitted fabrics using the knitted fabric parameters mass per unit area and thickness of knitted fabric. The regression model was implemented using WEKA 3.9.6 [18] on a dataset with 127 instances of two attributes, mass per unit and thickness of the fabric, taken from Table 2, and a class value for algorithm taken from Tables 3–5.

A comprehensive comparative analysis of nine regression models created in Weka was conducted. The developed models were: (1) random tree, (2) random forest, (3) linear regression, (4) multilayer perceptron, (5) support vector machine (SMOreg in Weka), (6) k-nearest neighbor classifier (IbK in Weka), (7) K* (KStar in Weka), (8) additive regression, and (9) regression by discretization.

These models were employed with their default parameter as set in WEKA. Model evaluation was conducted using 10-fold cross-validation, wherein the dataset was randomly divided into 10 equal-sized subsamples. One subsample was utilized for model validation, while the remaining nine were employed for training the model, following the principles outlined in [19]. Each experiment was iterated 10 times, each iteration employing a distinct random seed, resulting in a total of 100 results per model.

The results of the obtained correlation coefficients are presented in Table 7. Correlation coefficients close to 1 indicate a strong linear relationship between the predicted and actual values. The asterisk sign (*) in the superscript indicates the base model obtained a statistically higher result than the model it is compared to. The base model in this research is the random tree model. Similarly, the v sign in the superscript indicates the base model obtained a statistically lower result than the model it is compared to. For example, a correlation coefficient value of 0.95 obtained by the random tree model is statistically significantly higher than the one obtained by the linear regression model at 0.51, with significance $\alpha = 0.05$.

Table 7. Correlation coefficient obtained by machine learning models created in Weka across all samples. The superscript asterisk (*) serves as an indicator, denoting that the baseline model, specifically the random tree model, as shown in column (1), achieved a significantly higher outcome when compared to the model with the presence of an asterisk.

	Trees		Functions			Lazy		Meta	
Algorithm	**(1)**	**(2)**	**(3)**	**(4)**	**(5)**	**(6)**	**(7)**	**(8)**	**(9)**
ML-1	0.90	0.90	0.23 *	0.23 *	0.90	0.21 *	0.89	0.82	0.88
ML-2	0.94	0.94	0.52 *	0.82 *	0.94	0.53 *	0.93	0.87 *	0.91
MT-1	0.92	0.92	0.26 *	0.49 *	0.92	0.30 *	0.92	0.84 *	0.91
MT-2	0.95	0.94	0.55 *	0.82 *	0.95	0.55 *	0.94	0.85 *	0.92 *
ST	0.93	0.93	0.42 *	0.74 *	0.93	0.42 *	0.92	0.85 *	0.88 *

Linear regression and k-nearest neighbor classifier have comparatively lower correlations, while random tree, random forest, support vector machine, and K* show strong correlations. Although the latter four models might obtain results with similar accuracy and can be used for porosity prediction, the random tree model is shown in Figure 6 due to its simplicity and error analysis in Section 4.5.
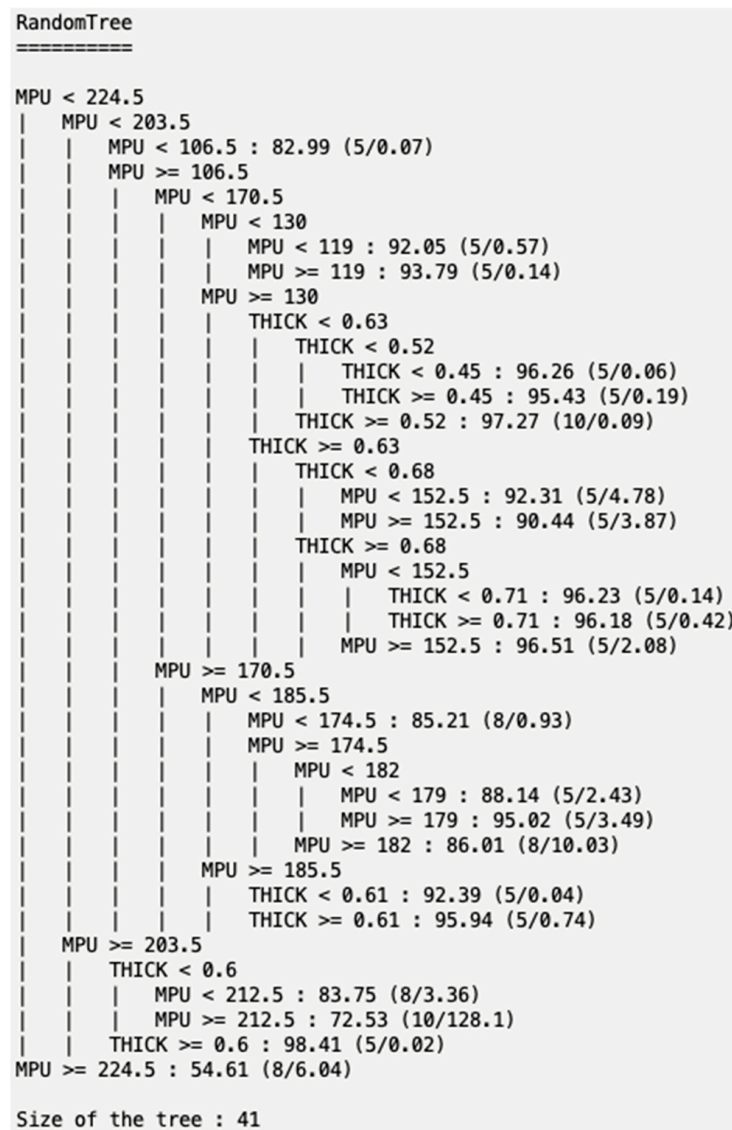
```
RandomTree
==========

MPU < 224.5
|   MPU < 203.5
|   |   MPU < 106.5 : 82.99 (5/0.07)
|   |   MPU >= 106.5
|   |   |   MPU < 170.5
|   |   |   |   MPU < 130
|   |   |   |   |   MPU < 119 : 92.05 (5/0.57)
|   |   |   |   |   MPU >= 119 : 93.79 (5/0.14)
|   |   |   |   MPU >= 130
|   |   |   |   |   THICK < 0.63
|   |   |   |   |   |   THICK < 0.52
|   |   |   |   |   |   |   THICK < 0.45 : 96.26 (5/0.06)
|   |   |   |   |   |   |   THICK >= 0.45 : 95.43 (5/0.19)
|   |   |   |   |   |   THICK >= 0.52 : 97.27 (10/0.09)
|   |   |   |   |   THICK >= 0.63
|   |   |   |   |   |   THICK < 0.68
|   |   |   |   |   |   |   MPU < 152.5 : 92.31 (5/4.78)
|   |   |   |   |   |   |   MPU >= 152.5 : 90.44 (5/3.87)
|   |   |   |   |   |   THICK >= 0.68
|   |   |   |   |   |   |   MPU < 152.5
|   |   |   |   |   |   |   |   THICK < 0.71 : 96.23 (5/0.14)
|   |   |   |   |   |   |   |   THICK >= 0.71 : 96.18 (5/0.42)
|   |   |   |   |   |   |   MPU >= 152.5 : 96.51 (5/2.08)
|   |   |   MPU >= 170.5
|   |   |   |   MPU < 185.5
|   |   |   |   |   MPU < 174.5 : 85.21 (8/0.93)
|   |   |   |   |   MPU >= 174.5
|   |   |   |   |   |   MPU < 182
|   |   |   |   |   |   |   MPU < 179 : 88.14 (5/2.43)
|   |   |   |   |   |   |   MPU >= 179 : 95.02 (5/3.49)
|   |   |   |   |   |   MPU >= 182 : 86.01 (8/10.03)
|   |   |   |   MPU >= 185.5
|   |   |   |   |   THICK < 0.61 : 92.39 (5/0.04)
|   |   |   |   |   THICK >= 0.61 : 95.94 (5/0.74)
|   MPU >= 203.5
|   |   THICK < 0.6
|   |   |   MPU < 212.5 : 83.75 (8/3.36)
|   |   |   MPU >= 212.5 : 72.53 (10/128.1)
|   |   THICK >= 0.6 : 98.41 (5/0.02)
MPU >= 224.5 : 54.61 (8/6.04)

Size of the tree : 41
```

Figure 6. Random tree model from Weka.

4. Discussion

In this research, five algorithms were developed for cover factor calculation: ST and improvement algorithms MT-1, MT-2, ML-1, and ML-2. In this section, we delve into the development and comparison of cover factor calculation algorithms. Notably, ML and MT differ in threshold determination and subsequent pseudo-grayscale transformations. In ML-1, the grayscale range is evenly divided into eight parts, employing empirically chosen thresholds (T1 to T8) for creating a pseudo-grayscale image. Contrastingly, MT-1 utilizes the Otsu method for threshold calculation, optimizing the separation of porous and non-porous regions. Both algorithms then undergo a pseudo-grayscale to binary conversion, enhancing segmentation accuracy and accounting for image variations. This refinement allows for a more precise differentiation between the background and the fabric regions, leading to better identification and calculation of the cover factor.

An additional attempt to account for image variations was made with ML-2 and MT-2 algorithms. Fabric images can exhibit variations in color, texture, and complexity. By defining additional threshold ranges (1–6 as black and 7–8 as white), we account for such variations and adapt the segmentation algorithm to better handle the specific characteristics of their fabric images. These improvements, grounded in empirical observations, fine-tune the segmentation process, enabling more precise porosity calculation than the ST algorithm.

4.1. Assessing the Influence of Multiple and Adaptive Thresholding: Comparative Analysis of ML-1 and MT-1 with ST

In this part of the study, the influence of the adaptive thresholding processing steps on the performance of ML-1, MT-1, and ST algorithms is investigated across all samples. The goal was to assess hypothesis H1 that algorithms ML-1 and MT-1 will obtain higher cover factors than the basic algorithm with a fixed threshold ST. Visual comparison is presented in Figure 5.

The statistical significance of the results was evaluated using the Wilcoxon signed-rank test, a non-parametric test for paired samples [20]. Like other non-parametric tests, the Wilcoxon signed-rank test does not make any assumptions about the distribution of the differences, specifically the absence of a normal distribution assumption. This test is typically used when analyzing pairs of measurements from the same subjects under two different conditions, or when there exists a one-to-one relationship between the two samples, with each value in one group corresponding to one value in the other group. The Wilcoxon signed-rank test calculates the comparison between the positive and negative ranks of the differences [21].

Null hypothesis of Wilcoxon signed-rank test claims there is no difference between the variables ML-1 [%] and ST [%]. Results indicated that there is a significant large difference between ST (median = 94.4, n = 127) and ML-1 (median = 96.4, n = 127), Z = 9.8, $p < 0.001$, r = 0.9, with significance level 0.95. The null hypothesis is thus rejected.

Results of the two-tailed Wilcoxon signed-rank test also indicated that there is a significant medium difference between MT-1 (median = 96, n = 127) and ST (median = 94.4, n = 127), Z = 9.8, $p < 0.001$, r = 0.9, with significance level 0.95. Null hypothesis that there is no difference between the variables MT-1 [%] and ST [%] is thus rejected.

Additionally, results of the Wilcoxon signed-rank test indicated that there is a significant medium difference between ML-1 (median = 96.4, n = 127) and MT-1 (median = 96, n = 127), Z = −4.8, $p < 0.001$, r = −0.4.

These findings suggest that utilizing the adaptive thresholding processing steps was effective in improving the performance of the algorithms for calculating porosity, thus making hypothesis H1 valid. Overall, the results underscore the importance of the adaptive thresholding processing steps in enhancing the accuracy and reliability of porosity calculations. The superiority of both ML-1 and MT-1 over ST highlights the potential of advanced algorithms and the validity of incorporating adaptive thresholding processing steps in porosity analysis.

4.2. Assessing the Impact of a Higher Threshold in ML-2 and MT-2 on Porosity Calculation: Comparison of ML-1 vs. ML-2 and MT-1 vs. MT-2 Algorithms

Accurately determining the cover factor of digital knitted fabric images presents its own set of challenges. One significant hurdle is the presence of stray yarns or fibers, posing difficulties in their distinction and introducing noise that can distort image analysis. These irregularities have the potential to impact cover factor precision, as the algorithm might misinterpret these stray elements. Additionally, variations in lighting conditions can cast shadows or shading, disrupting the perceived darkness or lightness of certain areas and significantly affecting the thresholding process. Recognizing these challenges, we pinpointed the selection of an appropriate threshold as a pivotal step in overcoming these issues. To tackle this, a larger threshold was integrated into the ML-1 and MT-1 algorithms, enhancing the accuracy of cover factor calculations.

As stated in Section 2, algorithm versions indicated with "−2" were created to further refine the porosity calculation by transforming pixels with pseudo-grayscale values from 1 to 6 and categorizing them as black, while those with a value of 7 or 8 were set to white. By categorizing pixels with specific pseudo-grayscale values as black or white, we aimed to better differentiate between the regions representing porous and non-porous areas in the image. This simplification of the pixel values could help create a clearer boundary between porosity and non-porosity, thereby improving the accuracy of the porosity calculation by

not including stray yarns that are present in the porosity holes due to their structure. In addressing these considerations and implementing necessary measures, our goal was to minimize the margin of error in determining the cover factor from digital yarn images.

In this section, we present the results of the statistical analysis of cover factor calculation algorithms: ML-1 vs. ML-2, and MT-1 vs. MT-2. The aim was to assess the validity of hypothesis H2. We test whether ML-2 and MT-2 will recognize the underlying knitted fabric structure and will obtain better results than their counterparts ML-1 and MT-1. Visual comparison is presented in Figure 5.

To determine whether there were significant differences between the paired samples, the non-parametric Wilcoxon signed-rank test was employed [20,22]. The null hypothesis assumed no difference between the first and second version of each ML or MT model, while the alternative hypothesis suggested a significant difference between them.

Results of the two-tailed Wilcoxon signed-rank test indicated that there is a significant large difference between ML-1 (median = 96.4, n = 127) and ML-2 (median = 92.4, n = 127), $Z = -9.8$, $p < 0.001$, r = -0.9 at $\alpha = 0.05$. Results also indicate that there is a significant large difference between MT-1 (median = 96, n = 127) and MT-2 (median = 89.5, n = 127), $Z = -9.8$, $p < 0.001$, r = -0.9, at $\alpha = 0.05$.

The comparison between ML2 and MT2 and the results of the Wilcoxon signed-rank test indicated that there is a significant large difference between ML-2 (median = 92.4, n = 127) and MT-2 (median = 89.5, n = 127), $Z = -7.3$, $p < 0.001$, r = -0.7, with significance level 0.95.

Additionally, since the ML-1 [%] group had higher values than the ML-2 [%] group, we performed a one-tailed Wilcoxon signed rank. The null hypothesis that the variable ML-1 [%] has a smaller or equal value compared to the variable ML-2 [%] was tested using a one-tailed t-test for paired samples. The result was statistically significant, W = 0, $p \leq 0.001$. The null hypothesis was rejected.

Lastly, the MT-1 [%] group had higher values than the MT-2 [%] group. The null hypothesis that the variable MT-1 [%] has a smaller or equal value compared to the variable MT-2 [%] was tested using a one-tailed t-test for paired samples. The result was statistically significant, at $\alpha = 0.05$, W = 0, $p \leq 0.001$. The null hypothesis is thus rejected, and hypothesis H2 has been confirmed.

When comparing the performance of two algorithms for porosity prediction, it is generally better to have a smaller porosity value, i.e., higher cover factor. A smaller porosity value indicates that there is less empty space and more solid fabric, which is often desired in various applications. However, despite the conventional preference for smaller porosity values, our research introduces a nuanced perspective. We believe that ML-2 and MT-2, despite yielding lower cover factor values (indicative of higher porosity) than ML-1 and MT-1, are anticipated to perform more effectively in capturing the true essence of fabric structure and to provide a more accurate representation of the actual fabric porosity, e.g., avoiding capturing stray yarns.

4.3. Analysis of Algorithmic Sensitivity to Cover Factor Calculations for Different Knitted Fabric Types

Fabrics designed for different purposes (football clothing—F, swimwear—S, and casual wear—I) inherently possess distinct structural features. These differences manifest in terms of cover factor and porosity, and algorithms capture these variations differently. Understanding the variations in cover factor across different fabric series is crucial for application-specific design, moisture management, and performance optimization. For example, certain sports demand specific fabric properties: swimwear benefits from quick-drying fabrics with high porosity while football clothing may require a balance between breathability and stretchability.

The primary objective of this study is to determine how sensitive each of the developed algorithms is to the specific fabric structure, by systematically comparing the cover factor values among distinct fabric series (Figure 7). By doing so, we will assess H3 that assumes

MT-1 algorithm, that uses Otzu adaptive thresholding method, will be the most sensitive algorithm in recognizing different fabric structures.
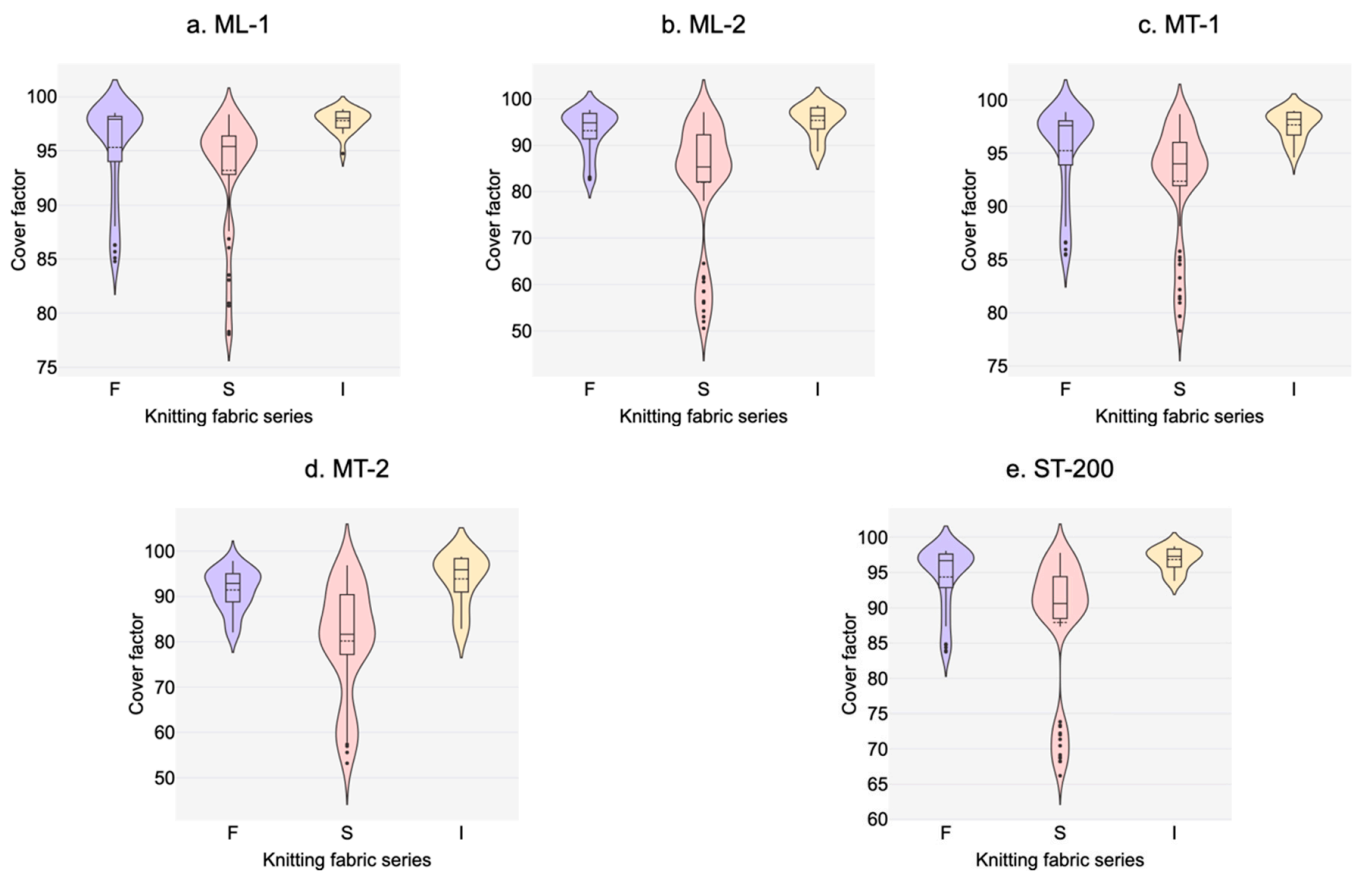

Figure 7. Violin plots in blue, red and yellow represent the distributions of the results of ML-1 algorithm (**a**), ML-2 algorithm (**b**), MT-1 algorithm (**c**), MT-2 algorithm (**d**), and ST algorithm (**e**) for samples in F, S and I series respectively.

Since F, S, and I are three distinct knitted fabrics series, the cover factor value should also be statistically significantly different, and a sufficiently sensitive algorithm should be able to distinguish the groups. Due to the distinct distribution for ML-2 and MT-2 observed in Figure 7, the hypothesis will be tested on the ML-1, MT-1, and ST algorithms.

To confirm this assumption, a non-parametric Kruskal–Wallis test was employed to assess the statistical difference of three fabric groups, F, S, and I, based on the cover factor results obtained by ML-1, MT-1, and ST [20,23]. Null hypothesis of the Kruskal–Wallis test is there is no difference between the three categories of the independent variable in terms of the dependent variable. The alternative hypothesis is there is a difference between the three categories of the independent variable in terms of the dependent variable.

For ML-1, the Kruskal–Wallis H test indicated that there is a significant difference in the dependent variable between the different groups, $\chi^2(2) = 29.82$, $p < 0.001$, with a mean rank score of 73.31 for F, 47.27 for S, 94.92 for I. The post-hoc Dunn's test using a Bonferroni corrected at $\alpha = 0.017$ indicated that the mean ranks of the following pairs are significantly different: F-S and S-I.

For MT-1, the Kruskal–Wallis H test indicated that there is a significant difference in the dependent variable between the different groups, $\chi^2(2) = 30.16$, $p < 0.001$, with a mean rank score of 71.82 for F, 47.71 for S, and 96.9 for I. The post-hoc Dunn's test using a Bonferroni corrected $\alpha = 0.017$ indicated that the mean ranks of the following pairs are significantly different: F-S, F-I and S-I.

For ST, the Kruskal–Wallis H test indicated that there is a significant difference in the dependent variable between the different groups, $\chi^2(2) = 37.8$, $p < 0.001$, with a mean rank score of 76.29 for F, 44.6 for S, 96.47 for I. The post-hoc Dunn's test using a Bonferroni corrected $\alpha = 0.017$ indicated that the mean ranks of the following pairs are significantly different: F-S and S-I.

The results of this study indicate that only the MT-1 algorithm can distinct the nuanced differences among knitted samples by recognizing series F, S, and I as statistically significantly different series, while ML-1 and ST recognize F-S and S-I as statistically significantly different series, therefore confirming H3.

Additional conclusions of this study are: (i) different algorithms interpret or quantify porosity in unique ways since they emphasize different aspects of the fabric structure, leading to variations in porosity calculations, (ii) some algorithms might be more sensitive to certain types of fabric structures or variations in porosity, and (iii) the yarn structure has an influence on the porosity levels observed in the fabric.

4.4. Comparison of Fixed Threshold vs. Adaptive Threshold Methods and Selection of Hyperparameter T

The choice of the threshold value T can significantly impact the performance of the algorithms. Using fixed threshold value for image segmentation has its benefits and drawbacks. One benefit of using a fixed threshold value, such as T = 200 in ST algorithm, is the simplicity it brings to the implementation. By choosing a specific threshold, the need for complex adaptive thresholding techniques is eliminated. This simplification makes the algorithms easier to understand and implement, reducing the complexity of the overall process.

Another advantage is the stability and reproducibility it offers in the results. When the same fixed threshold value is used across different datasets or experiments, the algorithms can produce consistent outcomes. This consistency facilitates easier comparison and analysis of the results, as researchers can rely on the stability of the threshold value.

However, there are also drawbacks to using fixed threshold values. One major drawback is the lack of adaptability. Fixed thresholds may not be optimal for all images or datasets. Different images can possess varying characteristics, such as differences in lighting, contrast, or noise levels. These variations can significantly affect the effectiveness of a fixed threshold. Consequently, a single threshold value may not be suitable for all situations, potentially leading to suboptimal segmentation results.

Furthermore, using a fixed threshold can make the algorithm sensitive to image variations. If the image quality changes or the characteristics of the objects being segmented vary, the chosen fixed threshold value may no longer be appropriate. This sensitivity to image variations can result in either over-segmentation (where objects are split into smaller regions) or under-segmentation (where objects are merged together), ultimately compromising the accuracy of the algorithm's segmentation results.

We have chosen the value of T = 200 based on empirical observations and experimentation. In Table 8, the results of the analysis of the samples using the simple threshold algorithm with threshold values other than 200 are listed for samples S4, S5, S7, and S9. In the case of sample S4, the contrast was not corrected and the threshold was set to T = 140. As for sample S5, the contrast was corrected but the threshold was set to T = 240. In the case of sample S7, the contrast was corrected and the threshold was set to T = 245. It is worth noting that photos 1 to 5 were taken before contrast enhancement, while photos 6 to 10 were taken after contrast enhancement, resulting in significant differences between these two subgroups. For the photos in sample S9, the contrast was corrected and the threshold was set to T = 240. The algorithm's performance with different threshold values for specific samples was subsequently investigated, and the results of the obtained cover factors using ST algorithm are presented in Table 8.

Table 8. Revised analysis for samples S4, S5, S7, and S9 for simple threshold algorithm with value threshold values different than T = 200.

Sample	ST [%]	Sample	ST [%]	Sample	ST [%]	Sample	ST [%]
S4-1	99.93	S5-1	95.39	S7-1	89.23	S9-1	97.92
S4-2	99.89	S5-2	96.16	S7-2	89.49	S9-2	97.69
S4-3	99.82	S5-3	93.96	S7-3	92.12	S9-3	97.67
S4-4	99.86	S5-4	93.69	S7-4	89.58	S9-4	97.78
S4-5	99.84	S5-5	95.44	S7-5	90.85	S9-5	97.70
S4-6	99.94	S5-6	96.05	S7-6	97.98	S9-6	97.40
S4-7	99.95	S5-7	95.90	S7-7	97.84	S9-7	97.63
S4-8	99.98	S5-8	96.19	S7-8	97.75	S9-8	97.61
				S7-9	97.91		
				S7-10	97.94		

In the case of sample S4, a lower threshold value of T = 140 was experimented with, due to the uncorrected contrast in the photographs of sample S4. By reducing the threshold, an attempt was made to compensate for the lower contrast and potentially improve the segmentation results. For sample S5, although the contrast was corrected, the threshold was increased to T = 240. The higher threshold was chosen based on the specific image characteristics of sample S5, which required a higher threshold to accurately distinguish between foreground and background regions. Similarly, for sample S7, the contrast was corrected, and the threshold was set to T = 245. The threshold value was chosen based on the image characteristics of sample S7, requiring a higher threshold due to the specific properties of the objects in the image. In contrast, for sample S9, with corrected contrast, the threshold value of T = 240 was reverted to. Threshold T = 240 was found suitable for achieving satisfactory segmentation results for this sample.

After evaluating the algorithm's performance on a representative set of images, it was determined that T = 200 yielded satisfactory results in terms of segmentation accuracy, computational efficiency, and ease of implementation.

However, we recognized the potential limitations of a fixed threshold and addressed that by developing algorithms that employ an adaptive thresholding technique (the Otsu method) to further expand on the ST algorithm. In this context, the simple threshold algorithm with T = 200 serves as a baseline for comparative purposes. By comparing the performance of the adaptive thresholding version (using Otsu's method) to the fixed threshold version, the effectiveness of the adaptive approach can be assessed and it can be determined whether it provides superior segmentation results.

4.5. Error Analysis and Comparative Performance Evaluation of Machine Learning Models

This section presents an in-depth error analysis of the computational models presented in Section 3.2. Since the primary focus is on understanding the correlation between various models and the actual porosity values of the knitted fabrics, the following metrics have been observed: the correlation coefficient across all models in Table 7, mean absolute error (MAE), root mean square error (RMSE), relative absolute error (RAE), and root relative squared error (RRSE) in Table 9.

Error analysis for algorithm MT-1 is presented in Table 9 because it has been proven to be the most sensitive in Section 4.3. Error analysis, however, has been performed on all 5 algorithms and they all support the conclusion in this section.

Table 9. Error analysis for algorithm MT-1. The superscript *v* sign (ᵛ) serves as an indicator, denoting that the baseline model, specifically the random tree model, as shown in column (1), achieved a significantly lower outcome when compared to the model with the presence of a *v* sign.

	Trees			Functions		Lazy		Meta	
Factors	**(1)**	**(2)**	**(3)**	**(4)**	**(5)**	**(6)**	**(7)**	**(8)**	**(9)**
Mean absolute error	1.20	1.21	3.68 ᵛ	3.67 ᵛ	1.20	3.31 ᵛ	1.37 ᵛ	1.99 ᵛ	1.34 ᵛ
Root mean square error	2.05	2.06	4.79 ᵛ	4.74 ᵛ	2.05	4.77 ᵛ	2.15	2.67 ᵛ	2.15
Relative absolute error	31.87	32.38	97.57 ᵛ	97.46 ᵛ	31.87	86.20 ᵛ	36.63 ᵛ	53.38 ᵛ	35.88 ᵛ
Root relative squared error	42.58	42.85	99.46 ᵛ	98.30 ᵛ	42.58	97.47 ᵛ	44.88	55.52 ᵛ	45.05

To determine the significance of differences between the predictive capabilities of the algorithms, a paired *t*-test was employed in Weka, with significance $\alpha = 0.05$. The random tree model served as a base model for statistical comparison shown in Table 9. The asterisk sign (*) in the superscript indicates the base model obtains statistically higher result than the model it is compared to. Similarly, the (ᵛ) sign in the superscript indicates the base model obtains statistically lower result than the model it is compared to.

Mean absolute error (MAE) represents the average absolute difference between predicted and actual values. Lower MAE values indicate better model performance. Random tree, random forest, and support vector machine have low MAE, suggesting better accuracy.

Root mean square error (RMSE) provides a measure of the average magnitude of errors. Lower values are desirable. Random tree, random forest, support vector machine, and K* show lower RMSE, indicating better predictive performance.

Similar to MAE and RMSE, lower values are better for both relative metrics. Random tree, random forest, k-nearest neighbor classifier, and K* exhibit lower relative errors.

In conclusion, random tree, random forest, and support vector machine models exhibit strong performance across multiple metrics with non-significantly different results, hence confirming hypothesis H4.

5. Conclusions

In this research we introduced and explored five algorithms—ST, MT-1, MT-2, ML-1, and ML-2—to study cover factor and porosity in knitted fabrics, with focus on knitted fabrics for sportswear, swimwear, and casual wear. We have also created computational models based on machine learning algorithms for cover factor prediction based on knitted fabric structural parameters.

The common steps for all the developed algorithms include grayscale conversion, contrast enhancement, thresholding, pixel counting, and cover factor calculation. The key difference between them is threshold selection. The single threshold algorithm (ST) relies on a fixed threshold, offering simplicity in its approach. In contrast, the multiple linear threshold algorithms (ML-1 and ML-2) and multiple threshold algorithms with the Otzu method (MT-1 and MT-2) introduce additional threshold ranges to account for variations in color, texture, and complexity in the pseudo-grayscale image conversion step. In ML algorithms, the grayscale range is divided linearly to perform pseudo-grayscale transformations with empirically chosen thresholds, while in the MT algorithms the Otsu method is used for threshold calculation, optimizing porous and non-porous region separation.

In algorithmic comparison, ML-1 consistently exhibits the highest average cover factor values. Comparing fixed (ST) and adaptive threshold (ML and MT) algorithms highlighted the advantages of the simplicity and stability of a fixed threshold. Yet, exploring alternative adaptive algorithms like Otsu's method suggests room for further refinement, acknowledging the limitations of fixed thresholds in adapting to diverse image characteristics. Our investigation into the impact of adaptive thresholding (ML-1 and MT-1) validated our hypothesis (H1) that these algorithms outperform the fixed threshold algorithm (ST). This

emphasizes the pivotal role of adaptive steps in beefing up the accuracy and reliability of porosity calculations.

Tackling challenges related to stray yarns and lighting variations, the use of a higher threshold in ML-2 and MT-2 demonstrated superior recognition of fabric structure, supporting H2. Despite these algorithms yielding lower cover factors, they proved more effective in capturing the true essence of fabric porosity by minimizing errors linked to stray elements.

Our sensitivity analysis across different fabric types (F, S, and I) affirmed H3, with MT-1 showing sensitivity and recognizing distinct fabric series as statistically significant. Sensitivity to fabric variations makes MT-1 a valuable tool for considering specific design aspects in textile manufacturing.

Lastly, computational regression models based on machine learning have been developed that are able to predict cover factor based on knitted fabric parameters with high accuracy. Random tree, random forest, and SMOreg models exhibit strong performance across multiple statistical metrics.

References

1. Ramratan; Kumar, R.; Sood, S. A Study on Porosity Related Aspects of Cotton Knitted Fabric with Single Jersey Structure for Improved Comfort Application for Garment. *JTEFT* **2020**, *6*, 199–204. [CrossRef]
2. Mezarcioz, S.; Ogulata, R.T. Modelling of Porosity in Knitted Fabrics. *J. Fash. Technol. Text. Eng.* **2015**, *s1*, 1–3. [CrossRef]
3. Havlová, M.; Špánková, J. Porosity of Knitted Fabrics in the Aspect of Air Permeability—Discussion of Selected Assumptions. *Fibres Text. East. Eur.* **2017**, *25*, 86–91. [CrossRef]
4. Benltoufa, S.; Fayala, F.; Cheikhrouhou, M.; Nasrallah, S.B. Porosity determination of jersey structure. *AUTEX Res. J.* **2007**, *7*, 7. [CrossRef]
5. Stipaničev, D. *Introduction to Digital Image Processing and Analysis*; University of Split: Split, Croatia, 1994.
6. Wardiningsih, W.; Troynikov, O. Influence of Cover Factor on Liquid Moisture Transport Performance of Bamboo Knitted Fabrics. *J. Text. Inst.* **2012**, *103*, 89–98. [CrossRef]
7. Tàpias, M.; Ralló, M.; Escofet, J.; Algaba, I.; Riva, A. Objective Measure of Woven Fabric's Cover Factor by Image Processing. *Text. Res. J.* **2010**, *80*, 35–44. [CrossRef]
8. Ragab, A.; Fouda, A.; El-Deeb, H.; Abou-Taleb, H. Determination of Pore Size, Porosity and Pore Size Distribution of Woven Structures by Image Analysis Techniques. *J. Text. Sci. Eng.* **2017**, *7*, 5. [CrossRef]
9. Imrith, M.K.; Unmar, R.; Rosunee, S. Determination of Knitted Fabric Porosity Using Digital Imaging Techniques. *Adv. Mater. Sci. Eng.* **2016**, *2016*, 1–10. [CrossRef]
10. Owczarek, M. The Impact and Importance of Fabric Image Preprocessing for the New Method of Individual Inter-Thread Pores Detection. *AUTEX Res. J.* **2020**, *20*, 250–262. [CrossRef]
11. *ISO 5084:1996*; Textiles—Determination of Thickness of Textiles and Textile Products. ISO Central Secretary, International Organization for Standardization: Geneva, Switzerland, 1996.
12. *MATLAB*, 8.6.0.267246 (R2015b); The MathWorks Inc.: Natick, MA, USA, 2015.

13. Gonzalez, R.C.; Woods, R.E. *Digital Image Processing*, 3rd ed.; Prentice Hall: Upper Saddle River, NJ, USA, 2008; ISBN 978-0-13-168728-8.

14. Otsu, N. A Threshold Selection Method from Gray-Level Histograms. *IEEE Trans. Syst. Man Cybern.* **1979**, *9*, 62–66. [CrossRef]

15. Bhavsar, P.; Safro, I.; Bouaynaya, N.; Polikar, R.; Dera, D. Chapter 12—Machine Learning in Transportation Data Analytics. In *Data Analytics for Intelligent Transportation Systems*; Chowdhury, M., Apon, A., Dey, K., Eds.; Elsevier: Amsterdam, The Netherlands, 2017; pp. 283–307. ISBN 978-0-12-809715-1.

16. Das, S.; Wahi, A. Digital Image Analysis Using Deep Learning Convolutional Neural Networks for Color Matching of Knitted Cotton Fabric. *J. Nat. Fibers* **2022**, *19*, 15716–15722. [CrossRef]

17. Fabijańska, A.; Jackowska-Strumiłło, L. Image Processing and Analysis Algorithms for Yarn Hairiness Determination. *Mach. Vis. Appl.* **2012**, *23*, 527–540. [CrossRef]

18. Frank, E.; Hall, M.A.; Witten, I.H. *The WEKA Workbench. Online Appendix for "Data Mining: Practical Machine Learning Tools and Techniques"*, 4th ed.; University of Waikato: Hamilton, New Zealand, 2006.

19. McCormick, C. K-Fold Cross-Validation, with MATLAB Code. Available online: http://mccormickml.com/2013/08/01/k-fold-cross-validation-with-matlab-code/ (accessed on 16 June 2019).

20. Parab, S.; Bhalerao, S. Choosing Statistical Test. *Int. J. Ayurveda Res.* **2010**, *1*, 187. [CrossRef] [PubMed]

21. Wilcoxon Signed Rank Test Calculator. Available online: https://www.statskingdom.com/175wilcoxon_signed_ranks.html (accessed on 28 November 2023).

22. Wilcoxon Signed-Rank Test Calculator. Available online: https://www.socscistatistics.com/tests/signedranks/default2.aspx (accessed on 28 November 2023).

23. Kruskal Wallis Test Calculator—With Post-Hoc Dunn's Test Multiple Comparisons. Available online: https://www.statskingdom.com/kruskal-wallis-calculator.html (accessed on 28 November 2023).