applied sciences

MDPI

Article

Comparative Study of Computational Methods for Classifying Red Blood Cell Elasticity

Hynek Bachratý, Peter Novotný *, Monika Smiešková, Katarína Bachratá and Samuel Molčan

Department of Software Technologies, Faculty of Management Science and Informatics, University of Žilina, 010 26 Žilina, Slovakia; hynek.bachraty@fri.uniza.sk (H.B.); monika.smieskova@fri.uniza.sk (M.S.); katarina.bachrata@fri.uniza.sk (K.B.); samuel.molcan@fri.uniza.sk (S.M.)

* Correspondence: peter.novotny@fri.uniza.sk

Abstract: The elasticity of red blood cells (RBCs) is crucial for their ability to fulfill their role in the blood. Decreased RBC deformability is associated with various pathological conditions. This study explores the application of machine learning to predict the elasticity of RBCs using both image data and detailed physical measurements derived from simulations. We simulated RBC behavior in a microfluidic channel. The simulation results provided the basis for generating data on which we applied machine learning techniques. We analyzed the surface-area-to-volume ratio of RBCs as an indicator of elasticity, employing statistical methods to differentiate between healthy and diseased RBCs. The Kolmogorov–Smirnov test confirmed significant differences between healthy and diseased RBCs, though distinctions among different types of diseased RBCs were less clear. We used decision tree models, including random forests and gradient boosting, to classify RBC elasticity based on predictors derived from simulation data. The comparison of the results with our previous work on deep neural networks shows improved classification accuracy in some scenarios. The study highlights the potential of machine learning to automate and enhance the analysis of RBC elasticity, with implications for clinical diagnostics.

Keywords: red blood cell; elasticity; machine learning; surface-area-to-volume ratio; decision trees

check for updates

Citation: Bachratý, H.; Novotný, P.; Smiešková, M.; Bachratá, K.; Molčan, S. Comparative Study of Computational Methods for Classifying Red Blood Cell Elasticity. *Appl. Sci.* **2024**, *14*, 9315. https://doi.org/10.3390/app14209315

Academic Editor: Anca Udristoiu

Received: 14 August 2024
Revised: 7 October 2024
Accepted: 9 October 2024
Published: 12 October 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

The elasticity of red blood cells (RBCs), also referred to as erythrocytes, is one of the key physiological parameters that affect their functionality and ability to perform their primary tasks, such as transporting oxygen and carbon dioxide between the lungs and tissues. They must be able to deform and pass through narrow splenic slits with a width of 1–2 μm. Erythrocytes that are unable to properly squeeze through these splenic slits become trapped and are removed from circulation [1,2]. A loss of RBC deformability accompanies many pathological conditions, such as diabetes, sickle cell anemia, thalassemia, malaria, and others [3–5]. Thus, RBC deformability is justified as a physical biomarker of RBC dysfunction. Another area where RBC deformability is important is transfusion medicine, as the loss of RBC deformability is a consequence of RBC aging and storage lesions [6].

The deformability of RBCs is determined by a combination of the viscoelasticity of the plasma membrane, the viscosity of the cytoplasm, and cell geometry [7–10]. These individual contributions are influenced by a wide range of endogenous and exogenous factors, including the composition and structure of the cell membrane, intracellular factors, physical and chemical environmental conditions, and various pathological states. A recent review study [11] provides a summary of the latest findings on these factors and highlights the need to analyze them in light of the new concept of eryptosis.

In the last decade, machine learning has been increasingly used in the biomedical field to analyze complex biological data and improve diagnostic procedures. In the context of determining the elasticity of RBCs, machine learning brings new possibilities for the analysis of large volumes of data, the identification of subtle patterns, and the prediction of the mechanical properties of RBCs.

One of the main uses of machine learning in determining the elasticity of RBCs is the analysis of image data obtained from microscopic measurements. Convolutional neural networks (CNNs) find use in the automatic analyses of the microscopic images of RBCs or for the segmentation of RBCs from complex images, which are crucial for subsequent analysis [12,13]. CNNs can identify and quantify the deformations of RBCs based on their shape and size, thus enabling the fast and accurate analysis of large datasets [14].

Machine learning methods are also used to predict the mechanical properties of RBCs based on various input data [15,16]. Various regression or classification models can be trained to predict RBC elasticity based on physiological and biochemical parameters such as ion concentrations, membrane composition, and cytoskeleton state. Deep neural networks can process complex data inputs and create predictive models that take into account non-linear relationships between various biological factors and RBC elasticity.

One of the great advantages of using machine learning is the automation of analyses. Machine learning algorithms can be integrated into diagnostic instruments, enabling the rapid and accurate analysis of RBC elasticity in a clinical setting. They can also effectively process and analyze big data, which is especially important for population studies and clinical research.

Although machine learning methods offer significant benefits, there are also some limitations and challenges. The most significant limitations include data quality, model interpretability, and overfitting. The accuracy of the models depends on the quality and quantity of input data. Poor-quality or incomplete data can lead to erroneous predictions. Some models, especially deep neural networks, can be difficult to interpret, complicating the understanding of biological mechanisms. And, in the absence of large enough datasets, models can be overtrained, which leads to weaker generalization to new, independent data.

Problems with data quality and range can be solved in standard ways (data augmentation, the identification and removal of outliers, etc.). We also used them in paper [17], where we investigated the RBC elasticity problem with CNN tools. The results are briefly mentioned in Section 3. In this study, we try other approaches and investigate whether (and, if so, under what circumstances) they can produce comparable results. We based this work on the same experiment and dataset as previously, but they are used differently and processed by other tools. The same starting point allows us to compare the obtained results more directly.

The first approach consists in monitoring the possibilities of obtaining similar results by using significantly different and (if possible) simpler data inputs and mathematical tools for their investigation. In Section 4, we use data on the surface and volume of moving RBCs, which are significantly reduced by basic statistical methods and then compared, with a focus on their elasticity. In Section 5, we directly focus on the RBC elasticity classification problem. We use the classic tool of decision trees for this, but, within it, the dependence of the range of used data and the accuracy of the obtained results are monitored in detail. The use of the rich output data of the simulation experiment provides us with a good opportunity for this.

The diagram in Figure 1 depicts the framework of our research. This study focuses on the part highlighted with a green background; the other part is discussed in [17].

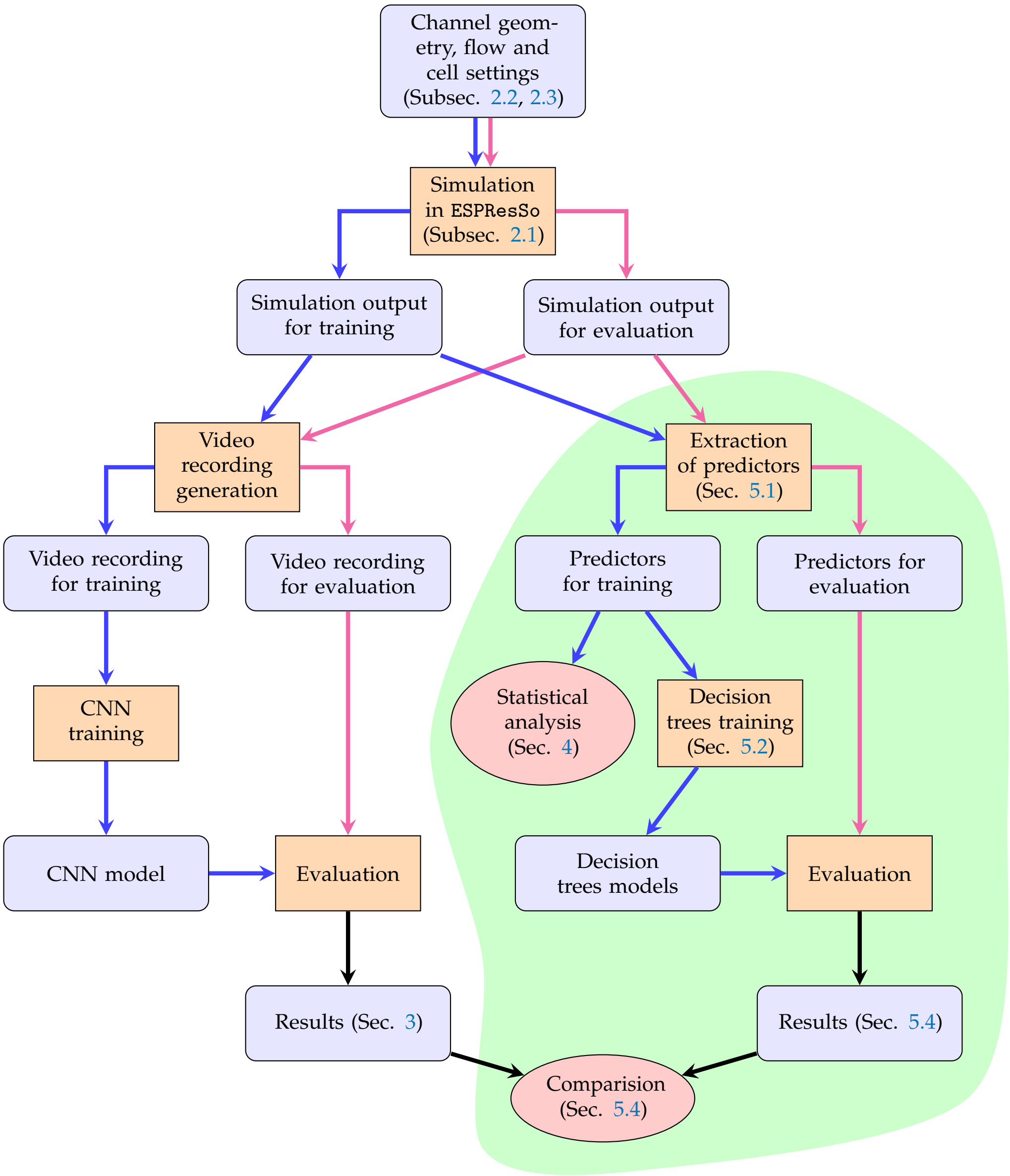
Channel geometry, flow and cell settings (Subsec. 2.2, 2.3)
Simulation in ESPResSo (Subsec. 2.1)
Simulation output for training
Simulation output for evaluation
Video recording generation
Extraction of predictors (Sec. 5.1)
Video recording for training
Video recording for evaluation
Predictors for training
Predictors for evaluation
CNN training
Statistical analysis (Sec. 4)
Decision trees training (Sec. 5.2)
CNN model
Evaluation
Decision trees models
Evaluation
Results (Sec. 3)
Results (Sec. 5.4)
Comparision (Sec. 5.4)

Figure 1. Summary of the framework. Blue and magenta arrows represent actions performed with training and evaluation data, respectively. Blue boxes represent input/output; orange boxes represent actions performed with the data.

2. Materials and Methods

2.1. Underlying Computational Model

The source data used to classify the elasticity of red blood cells were obtained from simulation experiments performed using the well-documented `PyOIF` module [18], which is part of the open-source software package `ESPResSo` [19] (version 4.1.2 was used). PyOIF uses a two-component model consisting of fluid and immersed elastic objects to model cell flow. The fluid dynamics are governed by the lattice Boltzmann method (LBM) [20] and the cells are represented using a spring mesh. Both components are linked via a dissipative version of the immersed boundary method (IBM) [21]. In addition to the fluid force, elastic forces act on the points of the cell mesh, which are evaluated from the deformation of the cell and repulsive forces originating from cell–cell and cell–channel interactions (obstacle or channel wall). At each time step, these forces are summed at each node of the cell membrane and used to propagate the node in space using Newton's second law of motion. The cells are created of the spring network that is formed by mesh points linked together by five elastic forces. These mesh points model the membrane of the cell, while the inside of the cell is filled with the same fluid as its surroundings. Elastic interactions, which are parameterized by elastic coefficients, model either the local elastic properties of the cell membrane (the stretching, bending, and local area interactions) or elastic properties related to the whole cell (the global volume and global area interactions).

2.2. Channel Geometry And Flow Setting

When RBCs come into contact with embedded obstacles, the deformability of RBCs is manifested to a greater extent. Therefore, the proposed microfluidic channel topology contains a periodic array of cylindrical obstacles, as in Figure 2. Blood cells have enough space to show their level of deformability during frequent interactions with obstacles. From the entire periodic array of obstacles, only a part consisting of five obstacles was simulated, while the periodic properties of the fluid were ensured in the direction of the fluid flow and in the direction of the channel width. The dimensions of the simulation box were 104 × 60 × 40 µm and the radius of the obstacles was 20 µm (see Figure 2).

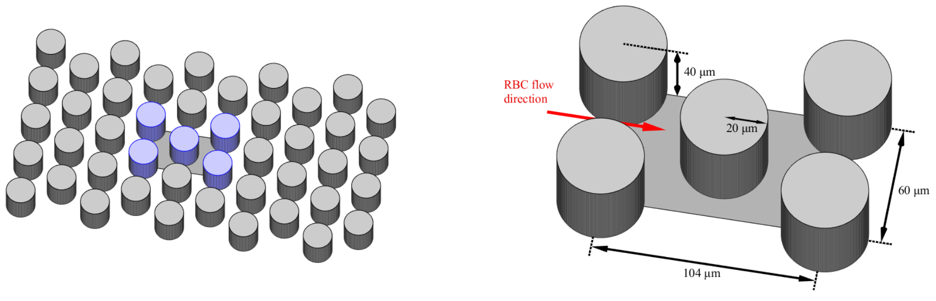

Figure 2. On the left, microfluidic channel topology is shown. Only the basic part with five obstacles (depicted with blue colour) was simulated. The figure on the right shows the scheme of the simulation box with the dimensions of the individual parts.

All simulations were performed with consistent channel and fluid flow parameters. They differed only in the initial seeding of cells, with cells placed in random positions in the simulation box. The fluid was discretized into a three-dimensional grid with an edge length of 1 µm. We are interested in studying flows in physically relevant cases; therefore, the density and viscosity values of the flowing fluid were 1025 kg/m^3 and 1.3 mPa s, similar to the values of blood plasma. External forces were used to initiate fluid flow, with values chosen to achieve a maximum velocity of approximately 0.03 m/s.

2.3. Cell Settings

RBCs were represented by a surface mesh consisting of 374 nodes. In the relaxed state, they had dimensions of 7.82 × 2.58 µm, and their volume and surface area were 92.7 µm^3 and 132.9 µm^2, respectively. In this study, four levels of RBC deformability were used, and each level was indicated by its respective value of the stretching coefficient, which has the greatest influence on deformability. All other coefficients had the same values at all levels of deformability. The values of the elastic coefficients of a healthy well-deformable RBC (denoted as $ks = 0.05$) were determined by calibration based on a biological experiment with optical tweezers [22]. More details about RBC validation can be found in [23]. The values of all the elastic coefficients of the healthy RBC are shown in Table 1. The least deformable RBCs showed rigidity approximately at the level of cells infected with malaria in the schizont stage (denoted as $ks = 0.03$). The elastic coefficients were chosen by calibration based on the results of experiments in [24]. The remaining two levels of RBC elasticity were equally distributed between healthy and infected RBCs with ks values of 0.0133 and 0.0216, respectively.

Each simulation involved 36 RBCs, representing 9 cells from each level of deformability. Mutual interactions between cells were modeled using the `membrane_collision` potential with the parameters `mc_K` $= 0.01$, `mc_n` $= 1.0$, `mc_cut` $= 0.4$. The interactions between the cells and the walls and obstacles were modeled using the `soft_sphere` potential with the parameters `soft_K` $= 0.001$, `soft_n` $= 1.2$, `soft_cut` $= 0.5$.

Table 1. Elastic coefficients of the healthy red blood cell (RBC) used in simulations.

Parameter	Value
stretching coefficient (k_s)	5×10^{-6} N/m
bending coefficient (k_b)	3×10^{-19} N/m
coefficient of local area conservation (k_{al})	2×10^{-5} N/m
coefficient of global area conservation (k_{ag})	7×10^{-4} N/m
coefficient of volume conservation (k_v)	900 N/m^2

3. Summary of Neural Network Results

In the previous study detailed in [17], we investigated the classification of RBCs from video recordings using convolutional neural networks, specifically employing neural networks with `ResNet` or `EfficientNet` architectures as the primary backbone. The video recordings were obtained from the simulation described above. The best accuracy was achieved with `EfficientNet_v2_B0` as the core model.

We categorized RBCs into four types based on their varying elasticity. Initially, for the task of four-class classification, we achieved an accuracy of 55.48%. However, upon subsequently reducing the target space to two classes, healthy and sick, we attained a significantly improved classification accuracy of 93.91%. Notably, this accuracy outperformed that of training neural networks directly for classifying RBCs into two classes, which had a final accuracy of 61.72%. The results are summarized in Table 2.

Table 2. Results obtained by neural networks with `EfficientNet_v2_B0` as the core model.

Training Number of Classes	Validation and Testing Number of Classes	Accuracy
4	4	55.48%
2	2	61.72%
4	2	93.91%

These findings led us to hypothesize that RBCs can be effectively segregated into healthy and diseased categories, with the distinction between individual types of diseased RBCs proving to be challenging. Our results suggest a division between healthy and diseased RBCs and the difficulty in distinguishing between specific types of diseased RBCs using current classification methodologies.

4. Statistical Analysis of Elasticity from the Perspective of Red Blood Cell Surface Optimization

One of the typical and key properties of RBCs is to achieve a shape that maximizes its surface for a given volume. We normally describe this value as the surface-area-to-volume ($SA : V$) ratio. Although elasticity is a dominant factor for other important properties of RBCs (the ability to penetrate the membrane in healthy RBCs; the clumping of RBCs in sickle-cell anemia and malaria), we were curious about the relationship, i.e., whether (and, if so, how) the different elasticities of RBCs would statistically manifest and differ in the values of the parameter $SA : V$. Articles [7,8] show that similar research and results can be obtained when studying the behavior of real RBCs. We investigated the relationship between RBC volume and surface area in our simulation model in [25].

4.1. Data and Analysis Tools

The basic data for comparison were calculated, and we recorded the surface area and volume values of individual RBCs in the simulation experiment. We are aware that the possibilities of obtaining these data from real experiments are limited, and the following analysis is methodologically suitable, especially for in silico experiments. Study [7], however, indicates the possibility of measuring $SA : V$ values also within in vitro experiments, including their connection to computational experiments. Study [8] also deals with the measurement of SA and V values.

We deliberately chose the statistical methods for further data processing to be as simple as possible in order to find out whether even such an approach will already bring relevant results. This simplicity should be somewhat of a counterpoint when compared to the latest machine learning methods.

4.2. Method of Analysis

As an output of the described simulation experiment, for each of the 36 cells with 4 different elasticity values, we used a sequence of 2356 values of its surface area and volume during the movement of RBCs in the simulation channel. We recorded the values after every 1000 simulation steps, which corresponded to a displacement of the RBC by an average of 1 to 2 micrometers in the x-axis direction of the channel. From each pair of SA and V values, we then calculated their $SA : V$ ratio and continued to work with these data.

The time series of the $SA : V$ parameter for each individual RBC generates interesting graphs (Figure 3), but statistically they are still quite large and confusing data.

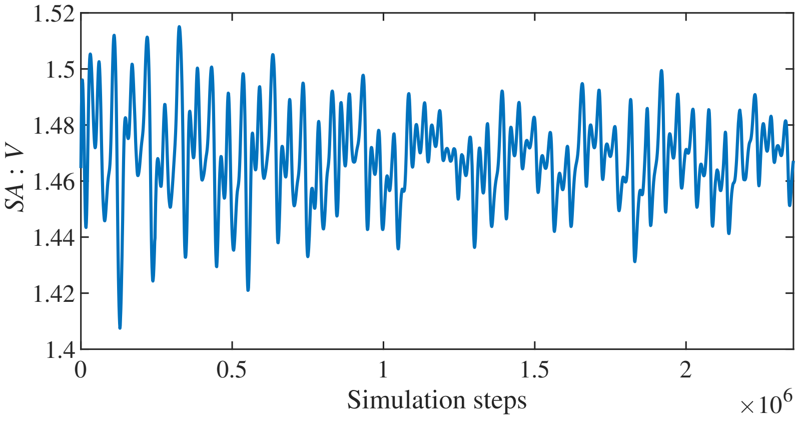

Figure 3. Time series plot of surface-area-to-volume ($SA : V$) ratio for a single healthy RBC.

We therefore decided to fundamentally reduce the data and, for each RBC from all 2356 values of the $SA : V$ ratio, we selected only their 4 basic characteristics: the maximum, minimum, average, and variance of $SA : V$ values. Figure 4 shows three of these four characteristics for all nine healthy RBCs.

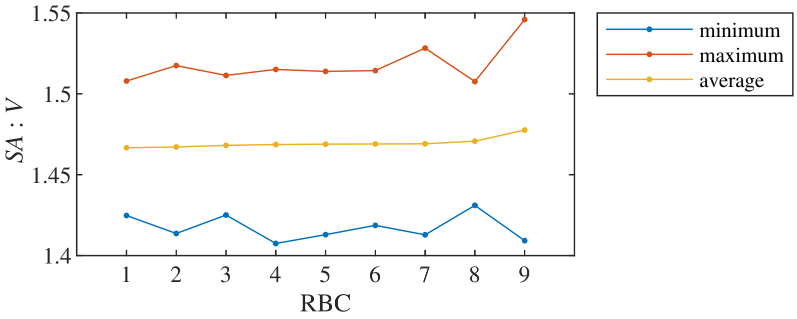

Figure 4. Minimum, maximum, and average $SA : V$ ratio for nine healthy RBCs. Cells are sorted by average.

For each statistic, we thus obtained four sets of nine values for the individual groups of RBCs with the same elasticity. Our next goal was to analyze whether it is possible to distinguish between these four sets based on these values.

4.3. Analysis Results

The combined display of the values of individual statistics for 4 × 9 RBCs in Figure 5 indicates that we obtain a good differentiation of healthy RBCs (type 0) from other types for the average and maximum values of $SA : V$. We do not see a difference for the minimum and variance values of $SA : V$, and we do not see a clear distinction between the three types of RBCs with reduced elasticity (type 1 to 3) in any of the statistics.

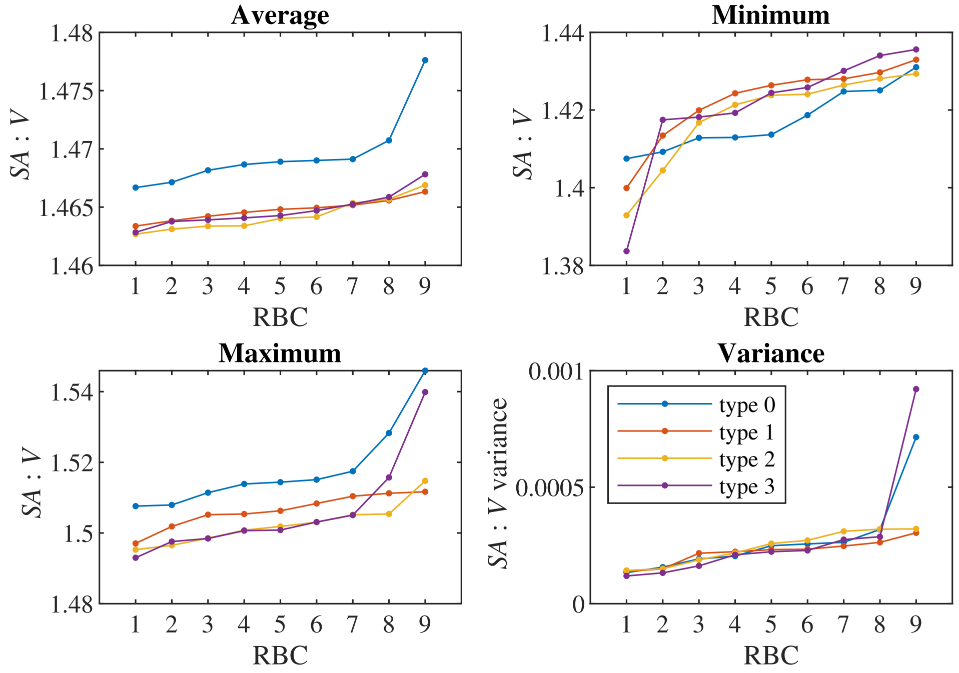

Figure 5. Average, minimum, maximum, and variance of $SA : V$ ratio for all 4 cell types. Cells of each type are sorted by the observed characteristic for each plot.

We verified these observations with the classical Kolmogorov–Smirnov (KS) test for the hypothesis that two given datasets come from the same (unspecified) probability distribution. The results are summarized in Tables 3–6. We reject the mentioned hypothesis for the values of averages and the values of maxima when comparing type 0 cells with the cells of types 1, 2, or 3 (green background). We can reject it at the standard level of significance of 5%; however, except for one case, the p-value is even less than 1%.

Table 3. Kolmogorov–Smirnov (KS) test results for average $SA : V$.

***p*-Value** ***D*-Value**	**Type 0**	**Type 1**	**Type 2**	**Type 3**
type 0		0	0.0007	0.0007
type 1	1		0.3517	0.9895
type 2	0.8889	0.4444		0.7301
type 3	0.8889	0.2222	0.3333	

Table 4. KS test results for minimum $SA : V$.

***p*-Value** ***D*-Value**	**Type 0**	**Type 1**	**Type 2**	**Type 3**
type 0		0.3517	0.7301	0.3517
type 1	0.4444		0.7301	0.9895
type 2	0.3333	0.3333		0.7301
type 3	0.4444	0.2222	0.3333	

Table 5. KS test results for maximum $SA : V$.

***p*-Value** ***D*-Value**	**Type 0**	**Type 1**	**Type 2**	**Type 3**
type 0		0.0336	0	0.0063
type 1	0.6667		0.1256	0.1259
type 2	1	0.5556		0.9895
type 3	0.7778	0.5556	0.2222	

Table 6. KS test results for the variance of $SA : V$.

***p*-Value** ***D*-Value**	**Type 0**	**Type 1**	**Type 2**	**Type 3**
type 0		0.7301	0.9895	0.9895
type 1	0.3333		0.7301	0.9895
type 2	0.2222	0.3333		0.9895
type 3	0.2222	0.2222	0.2222	

Thus, we can conclude that, by using the elementary statistics of maximum, minimum, average, and variance and the basic KS test, we can significantly differentiate the average and maximum values of the $SA : V$ ratio between healthy (type 0) and the other RBCs with reduced elasticity. We are not able to distinguish between types 1, 2, and 3 using these statistics.

We are aware of the limitation of the validity of our conclusion due to the small range of analyzed data given by the design of the simulation experiment. However, we consider the simplicity of the proposed methodology as a benefit since we can use it to verify the conclusions in a larger experiment in silico—and possibly also in vitro.

The confirmation of the result for the simulation experiments can also be expected based on the verified consistency and robustness of the computational model, when added RBCs, or a longer length of their simulated flow, should not cause significantly different behavior.

From the nature of the simulation model and the behavior of the real RBC, it is known that, during the described experiment or simulation run, i.e., during the flow of RBC through the channel, the RBC should maintain its volume, but its surface area may vary. We verified this assumption statistically (Figure 6) when the variance of the surface area values of simulated RBCs in the range $[0.5, 6]$ was approximately one hundred times greater than the variance of volume values in the range $[0.005, 0.06]$. Therefore, the obtained results would be essentially the same if we limited ourselves to examining only the SA values. This fact can be used to partially simplify the design of the necessary experiments or requirements for their output data.

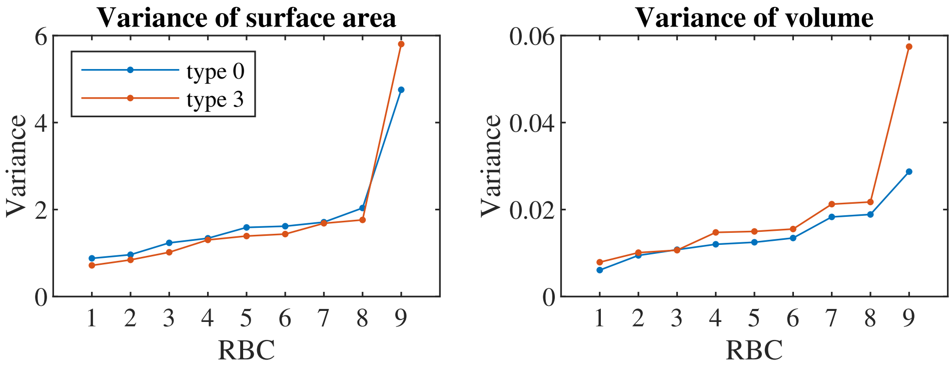

Figure 6. Variance of surface area and volume for cell types 0 and 3.

5. Classification Using Decision Trees

In [17], we dealt with the question of whether it is possible to classify RBC stiffness using deep learning based on analyzing the video recording of the flow of RBCs and what accuracy the classification can achieve in this way. Video recording is relatively easy to obtain. However, if we had more detailed data available on cell movement, which can be obtained, e.g., by recording the flow from several sides or using different sensors, it can be expected that the classification accuracy would be improved.

If we use video images directly for classification, it is natural to choose deep neural networks as the model. However, in the case where we have the specific properties of the captured cell available as data, describing its shape, speed, and changes in these properties over time, it makes sense to apply models that are suitable for working with tabular data. The simulation from which we generated the videos can be used as well to generate such data. Before investing in experiments with expensive sensors, we can thus verify the classification accuracy that can theoretically be expected with such a procedure. At the same time, we will be able to compare the classification ability of individual models and also evaluate which predictors have a significant impact on classification accuracy. This can be important when designing real experiments.

5.1. Used Predictors

We use cell triangulation to calculate predictors from the simulation. The output of the simulation for each cell is the position of each of the 374 triangulation nodes, determined by three coordinates in three-dimensional space. Various characteristics can be calculated from these data, which we subsequently use as predictors in classification. When designing the predictors, one should consider two aspects:

- How difficult it may be to extract the predictor from real experiments;
- Why one should expect that the elasticity of RBCs affects the given predictor.

In total, we created 41 predictors. We divided them into several sets according to the estimated complexity of obtaining them from a real experiment. Based on this, we created six tests, ordered according to the number of predictors we use. In the first test, we use only the most easily obtainable predictors, while, in the final test, we use all of them.

The following list describes the six predictor sets that we use in these tests (a comprehensive list of the predictors is included in Appendix A):

- Set 1: It consists of only two predictors—the dimensions of the rectangle in which the monitored cell is located, i.e., the dimensions of the cell in the direction of the x-axis and in the direction of the y-axis. These values can be easily obtained from a (static) snapshot of the cell. We may expect that the bounding box of an elastic RBC will vary in size more than the bounding box of a stiff RBC;
- Set 2: The additional predictors are velocities and changes in the velocity of the cell in the x- and y-directions. These data can be calculated from several consecutive images. The elasticity of an RBC affects its shape, which may affect the velocity of the cell in the flow;
- Set 3: It also includes dimensions and velocities in the z-axis direction. These values can be obtained from images taken from a different angle;
- Set 4: We added the cell axis length and the maximum and minimum diameter of the cell equator, which can potentially be determined from multi-angle images. While it is more difficult to obtain these characteristics from real experiments, one may expect that the shape of a cell is better represented by these values than just by the bounding box dimensions alone;
- Set 5: It additionally contains predictors that can be calculated from the complete cell triangulation—cell surface, cell volume and means, standard deviations and skewness coefficients calculated from the lengths of all triangulation edges, angles formed by every two triangulation triangles, and solid angles at all triangulation nodes. Cell triangulation is quite difficult to obtain from a real experiment; it would require the creation of a 3D image of the cell from the scanned flow. However, by using these complex characteristics, we may be able to distinguish more subtle changes in the shape of the RBC than with just the basic dimensions;

- Set 6: It also contains the means, standard deviations, and coefficients of the skewness of deviations and the absolute deviations of the characteristics added in the fifth set. The deviations are calculated from the cell in a relaxed state, so, for the calculation, we need to have the basic triangulation of the cell, which, in our procedure, corresponds to the cell in the first step of the simulation. One may expect that quantifying the changes of the shape represents the elasticity better than the original quantities themselves.

The mentioned predictors correspond to the current state of the cell at one monitored moment (in one step of the simulation). During experiments, however, we have the opportunity to observe the cell for longer during several steps. It can be expected that the classification accuracy will improve if we track the cell longer. To assess this effect, for each $S = 1, 10, 20, 40, 80, 160, 320, 640$, and 1280, we created a model in which we monitored each cell for $1000S$ simulation steps, with the simulation record saved once per 1000 steps. (Note that one pass of a cell through the channel corresponds to approximately 100,000 simulation steps.) Since the values of the individual predictors change during the movement of the cell through the channel, we have S values for each predictor instead of just one value. From these values, we calculated the mean and standard deviation and used them as predictors—in total, we have two predictors for each value described in the bulleted list above (with the exception of the case of $S = 1$, where we only work with static snapshots of the cells).

The technique to extract predictors from series data using statistics like mean, standard deviation, and skewness, described both in the previous paragraph and in the fifth and sixth predictor set, is a common approach [26–28], which was previously also used for RBC classification [29].

5.2. Used Machine Learning Tools

Given the nature of the data (number of predictors, high correlation of some predictors), we consider models based on decision trees, which usually achieve the best results for tabular data [30,31], to be a suitable tool for classification. We used two of the most popular techniques: random forest and gradient decision trees. For the implementation, we used the Python language (version 3.10.12) and the methods `ForestClassifier` from the `sklearn` library and `XGBClassifier` from the `xgboost` library.

The goal in this section was not to maximize the accuracy to the highest possible level; therefore, we did not optimize the hyperparameters of individual models and were satisfied with the default values.

For each of the six sets of predictors and each of the nine S values, we trained two models using both methods—one for classification into four classes (four types of cell stiffness), the other for classification into two classes (healthy/diseased cell). When evaluating the accuracy of the classification into two classes, in addition to the second model, we also used the first model, in which we combined the three types of stiffness into the result "diseased cell" (similarly, as we performed in [17]).

5.3. Data Preparation

We sampled the simulation for each recorded simulation step, after which followed at least $1000(S-1)$ additional steps (so that we could calculate predictors for a given value of S). However, we removed the first 100,000 steps of the simulation. The total number of generated samples is, thus, equal to $C \cdot (N - 100 - S + 1)$, where C is the total number of cells in the simulation and N is the total number of recorded simulation steps. (We remind you that every 1000th step is recorded, so the total number of steps is approx. 1000 N).

It follows from the above that many generated samples are very similar to each other. The reason is that the cell changes little between two consecutive steps; moreover, for $S > 1$, two samples recorded close to each other have a large part of the data from which we generate the predictors in common (the last $S-1$ recorded steps of the sample are identical to the first $S-1$ recorded steps of the following sample). This must be kept in mind when dividing the set into a possible training and validation or testing part—it is not appropriate to use data from one simulation in several parts and a new simulation must be used each time.

Due to the tree models used and the omission of hyperparameter optimization, we did not need the validation part, so we we only needed two simulations. We created training samples from the simulation with values of $C = 36$ and $N = 2356$; the total number of training samples is, thus, in the range from 81,216 (for $S = 1$) to 35,172 (for $S = 1280$). The second simulation, from which we generated test samples, has parameters $C = 36$ and $N = 2289$, so the number of samples is in a similar range (from 78,804 for $S = 1$ to 32,760 for $S = 1280$).

We recall that the simulations are the same as we used in [17], that is, there are 9 cells from each class among the 36 cells.

5.4. Classification Results

Let us first have a look at the classification accuracy based on the value of S, that is, let us analyze the effect of the number of steps that we monitor on each cell. From the plots in Figures 7 and 8, it can be concluded that increasing S has a positive effect up to the level around the values of $S = 160$ or 320, from which the classification accuracy ceases to improve continuously. Thus, it seems pointless to track the cell for significantly longer than one pass through the channel (approx. 100,000 simulation steps).

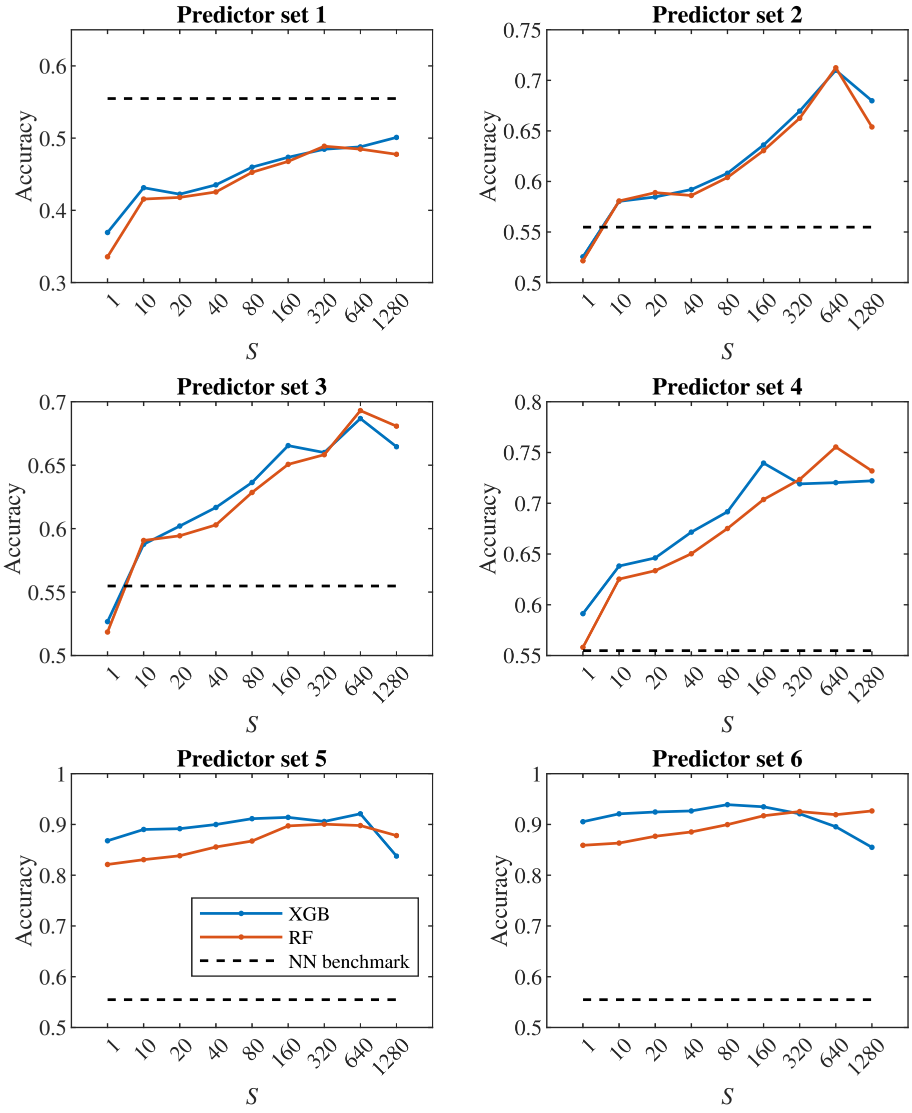

Figure 7. Dependence of classification results on S when predicting 4 classes.

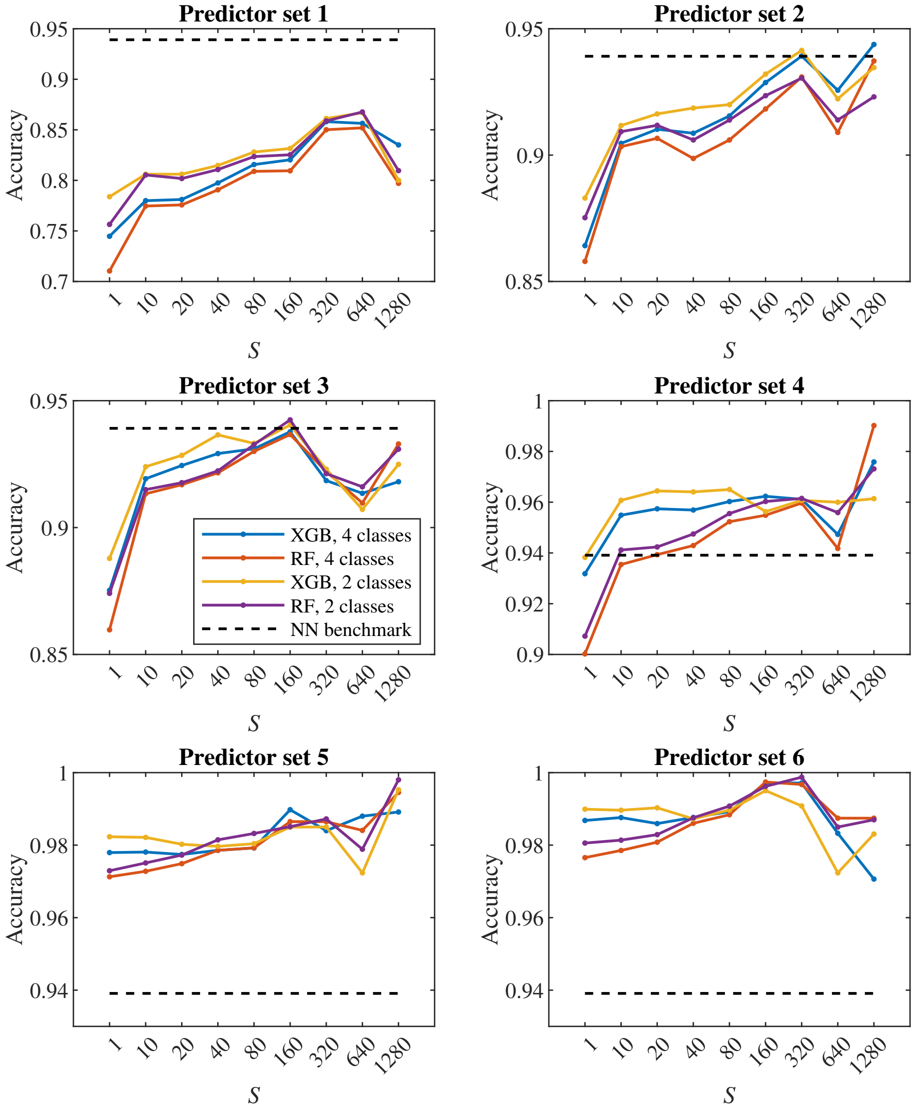

Figure 8. Dependence of classification results on S when predicting 2 classes.

The plots in Figures 9 and 10 show how the classification accuracy changes, depending on which set of predictors we use. Especially when classifying into four classes, a significant improvement in accuracy can be observed when moving from the fourth set to the fifth set. On the other side, the effect of the third dimension (the transition from the second set to the third set) does not seem to be very significant.

In all Figures 7–10, the dashed line shows the accuracy that we achieved in the classification using deep neural networks in [17]. Within that approach, we observed each cell during one passage through the channel, which corresponds approximately to a model with $S = 80$. In Figure 9 in the middle plot, it can be seen that, when classifying into four classes, we outperformed the neural networks' results with the second predictor set. Similarly, the middle plot in Figure 10 shows that, in the case of two classes, we outperformed neural networks with the fourth predictor set.

In [17], we observed a fairly significant difference in the result between a pair of models created for binary classification (healthy/diseased cell). The model that was trained directly for the classification of two classes achieved less accuracy (slightly above 60%) than the model trained for the classification of four classes, in which we determined the resulting binary classification only by merging the classes containing diseased cells with three different stiffness levels into one class (accuracy above 90%). This strange phenomenon did not appear when decision trees were used—in Figures 8 and 10, it can be seen that the accuracy of models trained for the classification of four classes (blue and red color) does not differ significantly from the accuracy of models trained directly for binary classification (yellow and purple color).

A comparison of random forest (RF) and gradient decision trees (XGB) does not come out significantly in favor of either method—as can be seen from Figures 7–10, the results are very similar to each other. One can notice that, in the case of classification into four classes (Figure 7), XGB outperforms RF on predictor sets 4–6 for $S \leq 160$, but its classification accuracy drops for $S \geq 320$. We believe the reason is that XGB is more prone to overfitting than RF and, for higher values of S, the data contain many samples that are very similar to one another, which may cause the model to fit to noise more easily than for smaller values of S with a greater variety of samples.

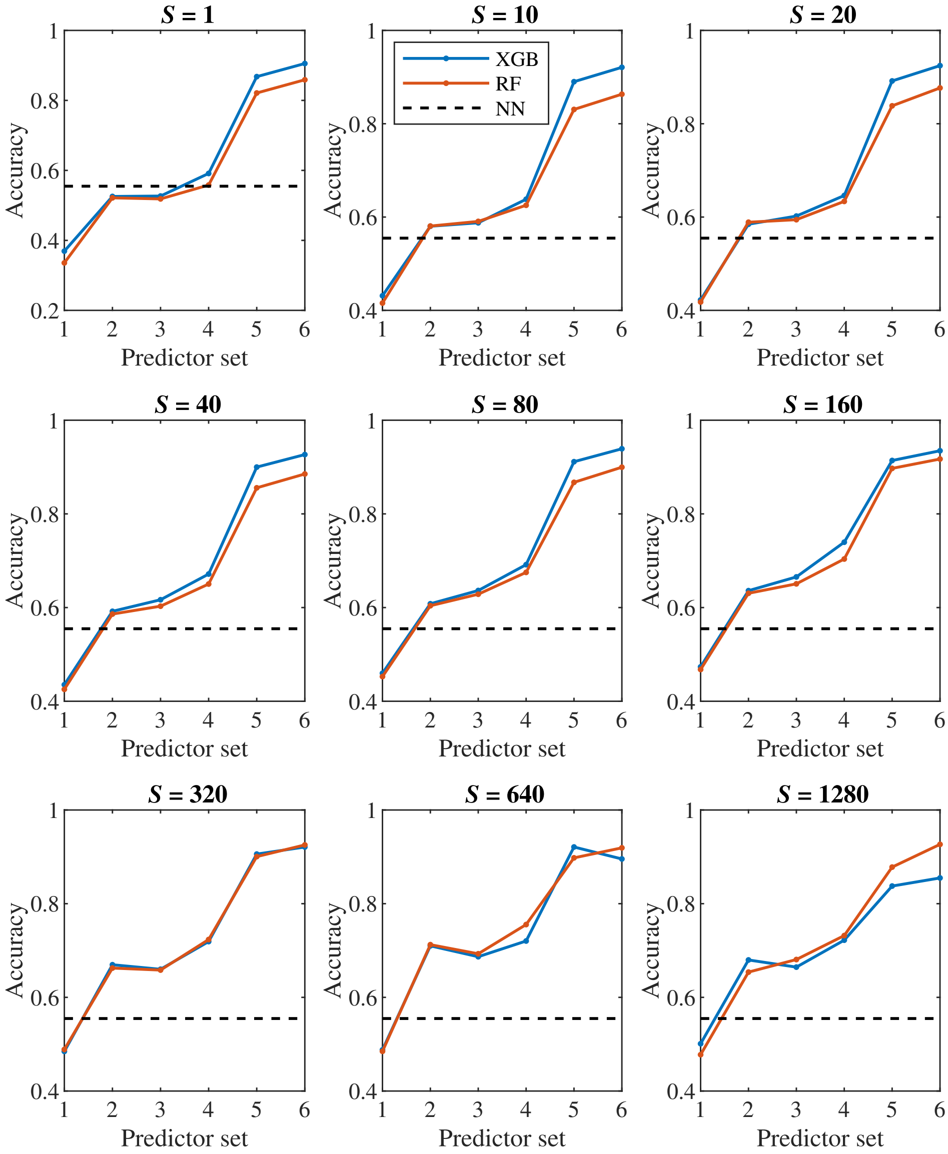

Figure 9. Dependence of classification results on predictor set when predicting 4 classes.

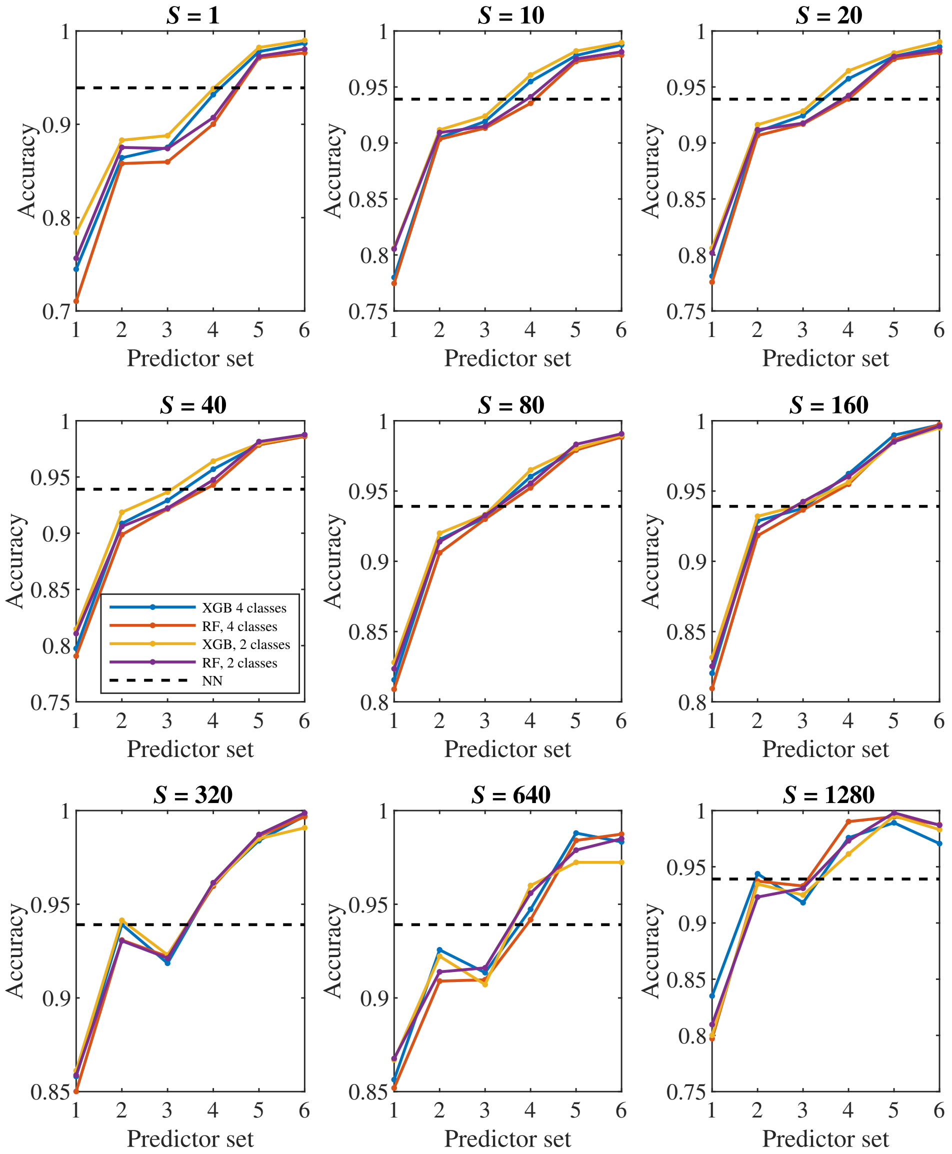

Figure 10. Dependence of classification results on predictor set when predicting 2 classes.

To see which predictors have the biggest impact on the classification, let us take a closer look at the models for $S = 80$ (i.e., those where we observe approximately one passage of a cell through the channel) with the sixth set of predictors and the fourth set of predictors. For these models in Figures 11 and 12, we present the significance of the predictors calculated by the `permutation_importance` method from the `sklearn` library. This is a technique used to evaluate the significance of each predictor in a trained machine learning model. This method rates how much each predictor contributes to the model's classification accuracy by measuring the change in the accuracy when the values of the single predictor are randomly shuffled along the entire data. It was first described in [32].

In the case of the sixth set and classification into four classes, the "edge length delta abs deviation" predictor appears to be the most significant in both models, that is, the predictor indicating the standard deviation of the absolute changes in the lengths of the edges of the triangulation against the relaxed state of the cell. When classifying into two classes, "edge angle delta abs mean" and "edge length delta skewness" are significant for both models, that is, the average absolute change in the angles at the edges of the triangulation compared to the relaxed state and the coefficient of the skewness of the changes in the lengths of the edges of the triangulation compared to the relaxed state. For the simpler fourth set of predictors, the most significant predictor for all models is the mean of the cell axis length. An explanation of the meaning of each predictor is given in Appendix A.

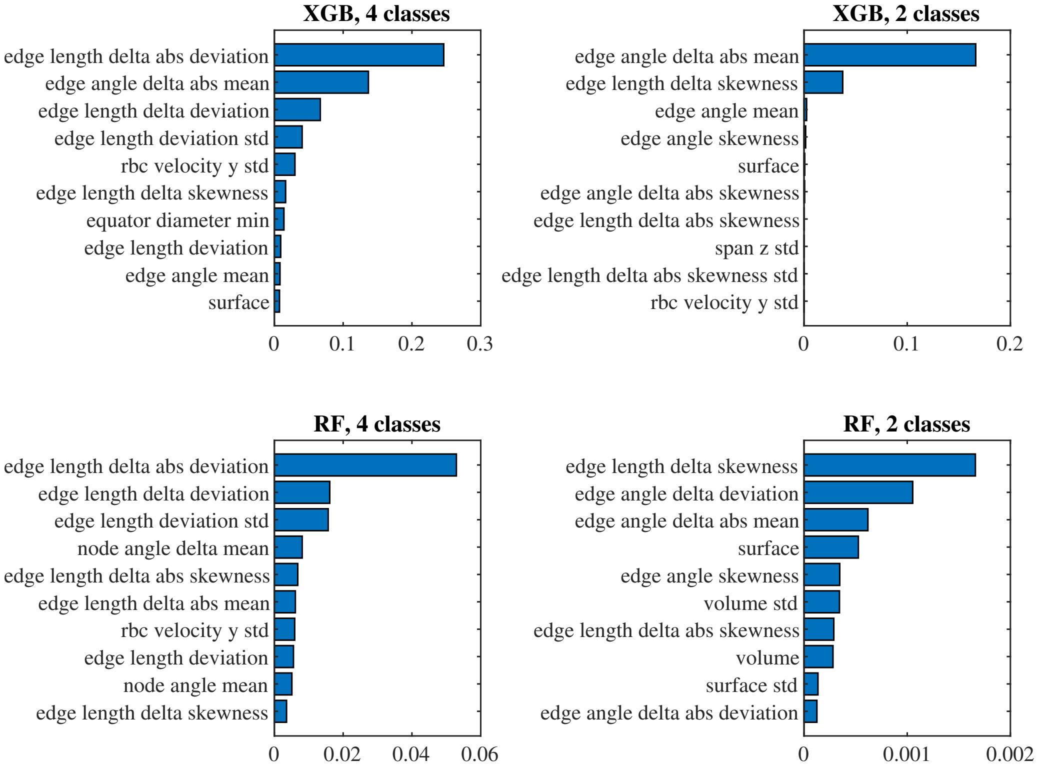

Figure 11. Importance of predictors from the 6th set.

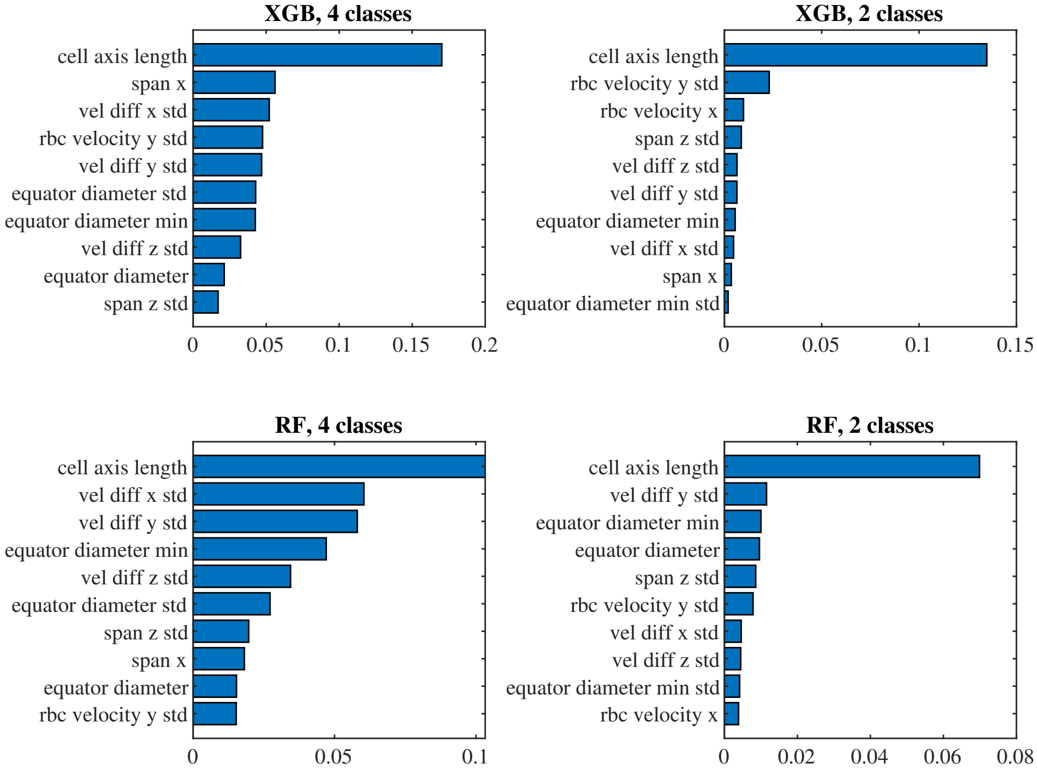

Figure 12. Importance of predictors from the 4th set.

6. Discussion

In Section 4, we focused on examining the elasticity of RBCs based on data on the $SA:V$ ratio during their flow through a channel. These data represent one of the fundamental characteristics of RBCs; they are frequently studied and can also be obtained from real experiments, not just from simulation. The data from the movement of each RBC were first reduced to four basic statistics: the maximum, minimum, average, and dispersion of the $SA:V$ ratio. The classical KS test revealed that the maximum and average values could differentiate healthy RBCs from a combined group of damaged ones based on their elasticity. However, it is not possible to differentiate between the three groups of damaged RBCs. The minimum and dispersion values do not distinguish between RBC groups at all using the chosen statistical methods. These findings are noted as statistical facts, even though it would be possible to look for different physical justifications for the specific design of the experiment.

In any case, we consider it worth noting that the possibility of distinguishing RBCs according to their elasticity found here qualitatively corresponds to the result obtained using the CNN mentioned in Section 3. Here, the classification of RBCs into four categories of elasticity was used when training the CNN, but the correct classification was achieved only for the group of healthy and aggregated group of damaged cells. Due to the fundamental difference in the tools used to achieve these results, a direct numerical comparison is not possible, but it inspires a further investigation of this phenomenon.

Our experiments with decision trees in Section 5 reveal insights into the efficiency of various predictor sets and the impact of recorded sequence length on the classification accuracy. Our results suggest that increasing the length of the sequence (S) generally improves classification accuracy up to a certain point, after which the improvements starts to be negligible. This indicates that, while longer observation periods can be beneficial, excessively long monitoring may not yield proportional gains in accuracy. Specifically, monitoring cells for approximately one complete passage through the channel appears sufficient for optimal performance.

The comparison between deep neural networks and decision tree-based models reveals that the latter can achieve comparable or better accuracy with simpler predictor sets. This is particularly evident in the classification into four classes, where decision trees outperformed neural networks using the second predictor set. This confirms that decision trees are a proper tool to effectively handle tabular data, providing a more interpretable and efficient alternative to neural networks. In addition to stating the numerical differences in classification accuracy, it is appropriate to note that the input data for both methods are different: the input for CNN was the video recording of the flow, while the input for decision trees was the predictors extracted from the simulation output. Video recording is based on 2D data, while predictor sets 3–6 benefit from 3D data. On the other hand, predictor sets 1 and 2 work with very simple 2D-data-based features, and one can expect that a CNN will be able to extract more information from the video. Table 7 shows the classification accuracy on the test dataset. In the case of decision tree methods, we list here only the models for $S = 80$, which approximately corresponds to one pass of a cell through the channel—that was also the case in video recordings used in CNN models.

Future research can focus on several directions to further improve the classification of RBC elasticity. One potential way is the integration of various data sources, such as combining video recordings with additional sensors capturing the mechanical properties or biochemical markers of RBCs. This could provide a richer dataset and better classification accuracy. Performing experimental validation using actual RBC samples will be essential to confirm the theoretical findings and translate them into practical diagnostic tools.

Table 7. Comparison of the classification accuracy of convolutional neural network (CNN) and decision tree-based models (with $S = 80$).

Training Number of Classes	Testing Number of Classes	Model	Predictors	Accuracy
4	4	CNN	video recordings	55.48%
		XGB	set 1	45.98%
		RF	set 1	45.27%
		XGB	set 2	60.81%
		RF	set 2	60.39%
		XGB	set 3	63.64%
		RF	set 3	62.85%
		XGB	set 4	69.16%
		RF	set 4	67.51%
		XGB	set 5	91.14%
		RF	set 5	86.73%
		XGB	set 6	93.90%
		RF	set 6	89.96%
2	2	CNN	video recordings	61.72%
		XGB	set 1	82.81%
		RF	set 1	82.35%
		XGB	set 2	92.00%
		RF	set 2	91.39%
		XGB	set 3	93.32%
		RF	set 3	93.27%
		XGB	set 4	96.50%
		RF	set 4	95.55%
		XGB	set 5	98.04%
		RF	set 5	98.32%
		XGB	set 6	98.97%
		RF	set 6	99.08%
4	2	CNN	video recordings	93.91%
		XGB	set 1	81.57%
		RF	set 1	80.89%
		XGB	set 2	91.55%
		RF	set 2	90.60%
		XGB	set 3	93.10%
		RF	set 3	93.00%
		XGB	set 4	96.03%
		RF	set 4	95.23%
		XGB	set 5	97.92%
		RF	set 5	97.93%
		XGB	set 6	98.92%
		RF	set 6	98.84%

Author Contributions: Conceptualization, K.B., H.B. and P.N.; methodology, H.B., P.N. and M.S.; software, S.M., P.N. and M.S.; validation, S.M. and P.N.; formal analysis, H.B. and P.N.; investigation, H.B., S.M. and P.N.; resources, M.S. and H.B.; data curation, S.M., P.N. and M.S.; writing—original draft preparation, H.B., S.M., P.N. and M.S.; writing—review and editing, H.B., K.B., P.N., S.M. and M.S.; visualization, S.M., P.N. and M.S.; supervision, K.B.; project administration, K.B.; funding acquisition, K.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Ministry of Education, Science, Research, and Sport of the Slovak Republic under contract No. VEGA 1/0369/22.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original data presented in the study are openly available on GitHub at https://github.com/nofto/RBC-Elasticity-Classification (accessed on 31 July 2024). However, the complete output from the simulation is too large for public sharing capabilities and is available on request from the corresponding author. The open-source version of the ESPResSo software package with the implemented PyOIF module used in this study is available in GitHub at https://github.com/icimrak/espresso/tree/python (accessed on 31 July 2024).

Acknowledgments: We would like to acknowledge the use of ChatGPT, a generative AI tool, in the preparation of this manuscript. As non-native English speakers, we utilized ChatGPT to assist in the better formulation of several phrases. It is important to note that AI was not employed in the creation of any substantive content within this manuscript. This use of AI is in accordance with MDPI's guidelines, ensuring full responsibility for the originality, validity, and integrity of the content presented.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

CNN	convolutional neural network
IBM	immersed boundary method
KS test	Kolmogorov–Smirnov test
LBM	lattice Boltzmann method
RBC	red blood cell
RF	random forest classifier from the `sklearn` library
XGB	gradient decision trees classifier from the `xgboost` library

Appendix A. List of Predictors

Table A1 presents 41 predictors used in Section 5. When referring to predictors for tests with $S > 1$, suffix "std" added to the predictor name indicates the standard deviation of all S values, while the absence of this suffix indicates the mean of all S values (see Figures 11 and 12).

Table A1. Predictors used in decision tree classification.

Set	Predictor	Description
1	span x	size of the bounding box of the cell in the direction of x-axis
	span y	size of the bounding box of the cell in the direction of y-axis
2	rbc velocity x	velocity of the cell in the direction of x-axis
	rbc velocity y	velocity of the cell in the direction of y-axis
	vel diff x	change of the velocity of the cell in the direction of x-axis
	vel diff y	change of the velocity of the cell in the direction of y-axis
3	rbc velocity z	velocity of the cell in the direction of z-axis
	span z	size of the bounding box of the cell in the direction of z-axis
	vel diff z	change of the velocity of the cell in the direction of z-axis
4	cell axis length	distance between the two nodes of the cell which form the central axis of the cell
	equator diameter	maximum distance between two opposite nodes on the perimeter of the cell
	equator diameter min	minimum distance between two opposite nodes on the perimeter of the cell
5	volume	cell volume
	surface	cell surface
	edge angle mean	mean of the angles between neighbouring triangles of the cell triangulation
	edge angle deviation	standard deviation of the angles between neighbouring triangles of the cell triangulation
	edge angle skewness	skewness coefficient of the angles between neighbouring triangles of the cell triangulation
	node angle mean	mean of the solid angles at the nodes of the cell triangulation
	node angle deviation	standard deviation of the solid angles at the nodes of the cell triangulation
	node angle skewness	skewness coefficient of the solid angles at the nodes of the cell triangulation
	edge length mean	mean of the lengths of the edges of the cell triangulation
	edge length deviation	standard deviation of the lengths of the edges of the cell triangulation
	edge length skewness	skewness coefficient of the lengths of the edges of the cell triangulation

Table A1. *Cont.*

Set	Predictor	Description
6	edge angle delta mean	as “edge angle mean”, calculated based on the deviations from the relaxed state
	edge angle delta deviation	as “edge angle deviation”, calculated based on the deviations from the relaxed state
	edge angle delta skewness	as “edge angle skewness”, calculated based on the deviations from the relaxed state
	node angle delta mean	as ‘node angle mean”, calculated based on the deviations from the relaxed state
	node angle delta deviation	as “node angle deviation”, calculated based on the deviations from the relaxed state
	node angle delta skewness	as "node angle skewness", calculated based on the deviations from the relaxed state
	edge length delta mean	as “edge length mean”, calculated based on the deviations from the relaxed state
	edge length delta deviation	as “edge length deviation”, calculated based on the deviations from the relaxed state
	edge length delta skewness	as “edge length skewness”, calculated based on the deviations from the relaxed state
	edge angle delta abs mean	as “edge angle delta mean”, based on the absolute deviations
	edge angle delta abs deviation	as “edge angle delta deviation”, based on the absolute deviations
	edge angle delta abs skewness	as “edge angle delta skewness”, based on the absolute deviations
	node angle delta abs mean	as “node angle delta mean”, based on the absolute deviations
	node angle delta abs deviation	as “node angle delta deviation”, based on the absolute deviations
	node angle delta abs skewness	as “node angle delta skewness”, based on the absolute deviations
	edge length delta abs mean	as “edge length delta mean”, based on the absolute deviations
	edge length delta abs deviation	as “edge length delta deviation”, based on the absolute deviations
	edge length delta abs skewness	as “edge length delta skewness”, based on the absolute deviations

References

1. Klei, T.R.L.; Dalimot, J.; Nota, B.; Veldthuis, M.; Mul, F.P.J.; Rademakers, T.; Hoogenboezem, M.; Nagelkerke, S.Q.; van Ijcken, W.F.J.; Oole, E; et al. Hemolysis in the spleen drives erythrocyte turnover. *Blood* **2020**, *136*, 1579–1589. [CrossRef]
2. Duez, J.; Holleran, J.P.; Ndour, P.A.; Pionneau, C.; Diakité, S.; Roussel, C.; Dussiot, M.; Amireault, P.; Avery, V.M.; Buffet, P.A. Mechanical clearance of red blood cells by the human spleen: Potential therapeutic applications of a biomimetic RBC filtration method. *Transfus. Clin. Biol.* **2015**, *22*, 151–157. [CrossRef]
3. Tyas, D.A.; Hartati, S.; Harjoko, A.; Ratnaningsih, T. Morphological, texture, and color feature analysis for erythrocyte classification in thalassemia cases. *IEEE Access* **2020**, *8*, 69849–69860. [CrossRef]
4. Alapan, Y.; Matsuyama, Y.; Little, J.A.; Gurkan, U.A. Dynamic deformability of sickle red blood cells in microphysiological flow. *Technology* **2016**, *4*, 71–79. [CrossRef]
5. Depond, M.; Henry, B.; Buffet, P.; Ndour, P.A. Methods to investigate the deformability of RBC during malaria. *Front. Physiol.* **2020**, *10*, 1613. [CrossRef]
6. Islamzada, E.; Matthews, K.; Guo, Q.; Santoso, A.T.; Duffy, S.P.; Scott, M.D.; Ma, H. Deformability based sorting of stored red blood cells reveals donor-dependent aging curves. *Lab Chip* **2020**, *20*, 226–235. [CrossRef]
7. Namvar, A.; Blanch, A.J.; Dixon, M.W.; Carmo, O.M.S.; Liu, B.; Tiash, S.; Looker, O.; Andrew, D.; Chan, L.; Tham, W.; et al. Surface area-to-volume Ratio, Not Cellular Viscoelasticity, Is the Major Determinant of Red Blood Cell Traversal through Small Channels. *Cell. Microbiol.* **2021**, *23*, e13270. [CrossRef]
8. Renoux, C.; Faivre, M.; Bessaa, A.; Da Costa, L.; Joly, P.; Gauthier, A.; Connes, P. Impact of Surface-area-to-volume Ratio, Internal Viscosity and Membrane Viscoelasticity on Red Blood Cell Deformability Measured in Isotonic Condition. *Sci. Rep.* **2019**, *9*, 6771. [CrossRef]
9. Park, H.; Lee, S.; Ji, M.; Kim, K.; Son, Y.; Jang, S.; Park, Y. Measuring cell surface area and deformability of individual human red blood cells over blood storage using quantitative phase imaging. *Sci. Rep.* **2016**, *6*, 34257. [CrossRef]
10. Diez-Silva, M.; Dao, M.; Han, J.; Lim, C.T.; Suresh, S. Shape and biomechanical characteristics of human red blood cells in health and disease. *MRS Bull.* **2010**, *35*, 382–388. [CrossRef]
11. Brun, J.F.; Varlet-Marie, E.; Myzia, J.; de Mauverger, E.R.; Pretorius, E. Metabolic influences modulating erythrocyte deformability and eryptosis. *Metabolites* **2022**, *12*, 4. [CrossRef]
12. Lamoureux, E.S.; Cheng, Y.; Islamzada, E.; Matthews, K.; Duffy, S.P.; Ma, H. Biophysical Profiling of Red Blood Cells from Thin-film Blood Smears using Deep Learning. *Heliyon* **2024**, *10*, e35276. [CrossRef]
13. Molina, A.; Rodellar, J.; Boldú, L.; Acevedo, A.; Alférez, S.; Merino, A. Automatic identification of malaria and other red blood cell inclusions using convolutional neural networks. *Comput. Biol. Med.* **2021**, *136*, 104680. [CrossRef]
14. Lopes, M.G.; Recktenwald, S.M.; Simionato, G.; Eichler, H.; Wagner, C.; Quint, S.; Kaestner, L. Big data in transfusion medicine and artificial intelligence analysis for red blood cell quality control. *Transfus. Med. Hemother.* **2023**, *50*, 163–173. [CrossRef]
15. Aliyu, H.A.; Sudirman, R.; Razak, M.A.A.; Abd Wahab, M.A. Red blood cell classification: Deep learning architecture versus support vector machine. In Proceedings of the 2nd International Conference on Biosignal Analysis, Processing and Systems (ICBAPS), Kuching, Malaysia, 24–26 July 2018; pp. 142–147. [CrossRef]
16. Singh, V.; Srivastava, V.; Mehta, D.S. Machine learning-based screening of red blood cells using quantitative phase imaging with micro-spectrocolorimetry. *Opt. Laser Technol.* **2020**, *124*, 105980. [CrossRef]
17. Molčan, S.; Smiešková, M.; Bachratý, H.; Bachratá, K.; Novotný, P. Classification of red blood cells using time-distributed convolutional neural networks from simulated videos. *Appl. Sci.* **2023**, *13*, 7967. [CrossRef]

18. Jančigová, I.; Kovalčíková, K.; Weeber, R.; Cimrák, I. PyOIF: Computational tool for modelling of multi-cell flows in complex geometries. *PLoS Comput. Biol.* **2020**, *16*, e1008249. [CrossRef]
19. Weik, F.; Weeber, R.; Szuttor, K.; Breitsprecher, K.; de Graaf, J.; Kuron, M.; Landsgesell, J.; Menke, H.; Sean, D.; Holm, C. ESPResSo 4.0—An extensible software package for simulating soft matter systems. *Eur. Phys. J. Spec. Top.* **2019**, *227*, 1789–1816. [CrossRef]
20. Krüger, T.; Kusumaatmaja, H.; Kuzmin, A.; Shardt, O.; Silva, G.; Viggen, E.M. *The Lattice Boltzmann Method*, Springer International Publishing: Cham, Switzerland, 2017. [CrossRef]
21. Bušík, M.; Slavík, M.; Cimrák, I. Dissipative coupling of fluid and immersed objects for modelling of cells in flow. *Comput. Math. Methods Med.* **2018**, *2018*, 7842857. [CrossRef]
22. Mills, J.P.; Qie, L.; Dao, M.; Lim, C.T.; Suresh, S. Nonlinear elastic and viscoelastic deformation of the human red blood cell with optical tweezers. *MCB Mol. Cell. Biomech.* **2004**, *1*, 169. [CrossRef]
23. Jančigová, I.; Kovalčíková, K.; Bohiniková, A.; Cimrák, I. Spring-network model of red blood cell: From membrane mechanics to validation. *Int. J. Numer. Methods Fluids* **2020**, *92*, 1368–1393. [CrossRef]
24. Suresh, S.; Spatz, J.; Mills, J.P.; Micoulet, A.; Dao, M.; Lim, C.T., Beil, M.; Seufferlein, T. Connections between single-cell biomechanics and human disease states: gastrointestinal cancer and malaria. *Acta Biomater.* **2005**, *1*, 15–30. [CrossRef]
25. Cimrák, I.; Bachratá, K.; Bachratý, H.; Jančigová, I.; Tóthová, R.; Bušík, M.; Slavík, M.; Gusenbauer, M. Object-in-fluid framework in modeling of blood flow in microfluidic channels. *Commun.-Sci. Lett. Univ. Zilina* **2016**, *18*, 13–20. [CrossRef]
26. Brunello, A.; Marzano, E.; Montanari, A.; Sciavicco, G. J48SS: A novel decision tree approach for the handling of sequential and time series data. *Computers* **2019**, *8*, 21. [CrossRef]
27. Mutlag, W.K.; Ali, S.K.; Aydam, Z.M.; Taher, B.H. Feature extraction methods: a review. *J. Phys. Conf. Ser.* **2020**, *1591*, 012028. [CrossRef]
28. Rady, E.H.A.; Fawzy, H.; Fattah, A.M.A. Time Series Forecasting Using Tree Based Methods. *J. Stat. Appl. Probab.* **2021**, *10*, 229–244. [CrossRef]
29. Veluchamy, M.; Perumal, K.; Ponuchamy, T. Feature extraction and classification of blood cells using artificial neural network. *Am. J. Appl. Sci.* **2012**, *9*, 615. [CrossRef]
30. Shwartz-Ziv, R.; Armon, A. Tabular data: Deep learning is not all you need. *Inf. Fusion* **2022**, *81*, 84–90. [CrossRef]
31. Uddin, S.; Lu, H. Confirming the statistically significant superiority of tree-based machine learning algorithms over their counterparts for tabular data. *PLoS ONE* **2024**, *19*, e0301541. [CrossRef]
32. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.