sensors

MDPI

Article

On the Problem of State Recognition in Injection Molding Based on Accelerometer Data Sets

Julian Brunthaler [1,*,†], Patryk Grabski [1,†], Valentin Sturm [2,†], Wolfgang Lubowski [1] and Dmitry Efrosinin [3]

[1] AISEMO GmbH, 4675 Weibern, Austria
[2] Linz Center of Meachtronics GmbH, 4040 Linz, Austria
[3] Institute of Stochastics, Johannes Kepler University Linz, 4040 Linz, Austria
[*] Correspondence: julian.brunthaler@aisemo.com
[†] These authors contributed equally to this work.

Abstract: The last few decades have been characterised by a very active application of smart technologies in various fields of industry. This paper deals with industrial activities, such as injection molding, where it is required to monitor continuously the manufacturing process to identify both the effective running time and down-time periods. Supervised machine learning algorithms are developed to recognize automatically the periods of the injection molding machines. The former algorithm uses directly the features of the descriptive statistics, while the latter one utilizes a convolutional neural network. The automatic state recognition system is equipped with an 3D-accelerometer sensor whose datasets are used to train and verify the proposed algorithms. The novelty of our contribution is that accelerometer data-based machine learning models are used to distinguish producing and non-producing periods by means of recognition of key steps in an injection molding cycle. The first testing results show the approximate overall balanced accuracy of 72–92% that illustrates the large potential of the monitoring system with the accelerometer. According to the ANOVA test, there are no sufficient statistical differences between the comparative algorithms, but the results of the neural network exhibit higher variances of the defined accuracy metrics.

Keywords: injection molding; accelerometer sensor; state recognition; machine-learning algorithm; convolutional neural network.

check for updates

Citation: Brunthaler, J.; Grabski, P.; Sturm, V.; Lubowski, W.; Efrosinin, D. On the Problem of State Recognition in Injection Molding Based on Accelerometer Data Sets. *Sensors* **2022**, *22*, 6165. https://doi.org/10.3390/s22166165

Academic Editor: Vladimir M. Vishnevsky and Arkady Zaslavsky

Received: 6 July 2022
Accepted: 11 August 2022
Published: 17 August 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

1. Introduction

In the last few decades, there has been an increasing interest in the application of big data and smart technologies in industry to guarantee the specified production quality and efficiency, as well as to track various anomalies during a manufacturing process. Injection molding production also requires state-of-the-art technology for process monitoring, data collection and their classification. Injection molding is known to be a low-cost, high-capacity production technology in the plastics industry. It is usually a process which involves injecting molten polymer materials from a heated barrel cylinder into a closed mold, followed by cooling of the plastic product and then pushing it out from the mold. Although the process is fully automated, there is a need for some auxiliary monitoring system to monitor the production process with the aim to increase the efficiency and to reduce costs.

Despite some advances in injection molding production quality control, no universal automatic methods have yet been proposed to recognize real running-time and down-time periods. The ability to track this information would obviously improve the economic and environmental efficiency of this type of production. We develop the monitoring system equipped with a 3D-accelerometer sensor on the injection molding machine for this purpose. The time-ordered datasets are recorded by the sensor in the form of three-dimensional vectors. Then, they are used for time series classification to recognize such individual states as mold closing, injection unit actions, plasticizing and demolding. In contrast to the previous results, our analysis includes the following contributions. We develop two state recognition algorithms based on supervised paradigms. In the first case, the relatively small amount of labeled data are used to estimate threshold levels for the specified features of descriptive statistics. In the second case, a convolutional neural network is implemented directly to the datasets recorded by a monitoring system. The real data are used for algorithm training and verification. Different accuracy metrics of the classification models are evaluated. To determine whether one learning algorithm outperforms another, the ANOVA-test is used. In a post-processing procedure, it is determined whether the sequence of recognized cycle steps represents a correct injection molding cycle. This information is used to determine whether or not production is to take place. This post-processing procedure is not the subject of this publication. This publication deals only with the recognition of cycle steps in acceleration data.

The remaining chapters of this paper are structured as follows: first, the related work is presented in Section 2. Section 3 describes which data are analyzed and how they are obtained. After the specification of the pre-processing, the two classification algorithms presented in this paper—a tree based approach and an artificial neural network model—are described in Section 4. More information about the metrics that are used in the numerical analysis of these two models as well as the evaluation itself can be found in Section 5. Finally, the outcome is summarized and discussed in Section 6.

2. Related Work

Multiple publications discuss the topic of sensor technology and machine learning to support injection mold manufacturing. A main topic concerned in literature is the quality prediction and assessment using machine learning techniques; see, for instance, [1–4]. Another useful approach is the utilization of a simulation model or digital twin and machine learning to support manufacturing processes as presented in [5–7].
In recent years, multiple schemes employing sensor data for production support and quality control were proposed in literature, such as pressure [8–12] and temperature [8–11] or ultra-sonic measurements [13]. A review on different sensor types utilized in-mold for process control can be found in [14].

In [15], the authors present a comprehensive review on the application of multiple supervised and unsupervised machine learning techniques for monitoring injection mold processes. An assistance system for quality assurance in the plastic injection molding process was described in [16]. They used pressure, force and temperature sensors which were placed inside the tool cavity. A support vector machine was then used for the classification into good and bad parts. The authors of [17] presented an approach to include machine learning in an injection molding process with the goal of enhancing the production performance. In their study, they used a wide range of sensors attached to the tool. With these measurements, a machine learning model was trained to predict the best process parameters for the injection molding machine. Several machine learning algorithms were utilized in [18] to test and compare performances in quality prediction by means of main features of time and temperature by injection molding. The problem of the quality prediction was analysed in [19], where convolutional neural networks were trained on thermographic images. Among others, we can mention also the works [20,21], where the quality classification and anomaly detection in injection molding were discussed in the framework of different aspects.

Accelerometers have long been used to track the state of moving objects; see, for example, [22] and the literature overview with references therein, where the low frequency acceleration data were used for feature selection and classification in a human activity recognition problem. The authors used 50 Hz sensor data from a triaxial acceleration sensor; in addition, they also used data from a gyroscope sensor. From these data, they calculated 585 features, which where then scored by three different feature selection methods. To get the optimal feature set a classifier was trained several times with different number of features. The results show that about 40 features are enough to get the best classification performance.

To the best of our knowledge, there are very few papers dedicated to utilization of accelerometers to monitor the injection molding process. The authors of [23] proposed a monitoring scheme for an injection molding tool, which consists of a pressure and acceleration sensor directly integrated into the injection tool. The authors conclude that "a reduction in the maintenance costs of the injection tool is expected". The monitoring system proposed in paper [24] uses an accelerometer to record injection molding vibration signals. These values were then used as input variables for logistic modelling to predict the mold flash, which is an excess plastic that forms on the surface of injection molded parts.

3. Data

This paper is based on the recording of acceleration data from injection molding machines as they occur in a plastic manufacturing facility. The left part of Figure 1 shows such an injection molding machine, which was also used for the purpose of data collection for this work. The data are measured by an triaxial accelerometer with a sampling frequency of 100 Hz, which means that a sample is recorded every 10 ms. The sensor is permanently mounted on the injection molding machine and records the acceleration for each of the three mutually orthogonal x, y and z-axes. To guarantee reliable data recording, it is attached in a tamper-proof manner so that slippage is not possible, as shown on the right in Figure 1 where the sensor is mounted on the machine. On the right side of Figure 1, one can see the mounted acceleromoter. The left picture shows the whole injection molding machine, and the right circle shows the position of the sensor.

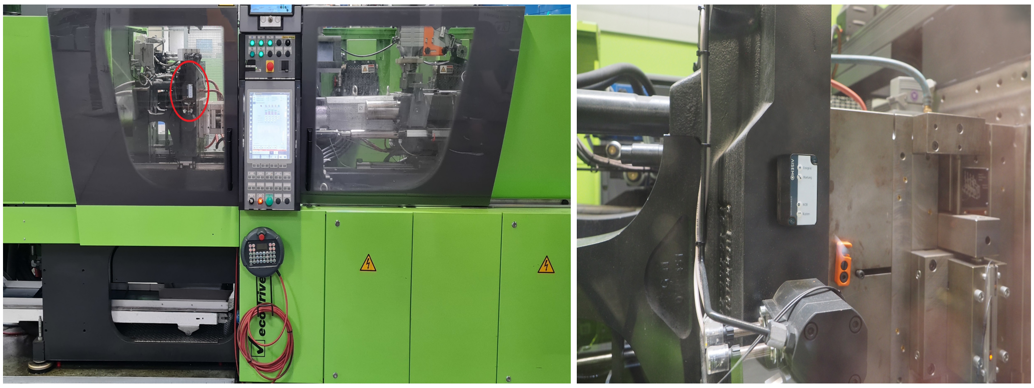

Figure 1. The left image shows an injection molding machine with a mounted 3-axis accelerometer (sensor marked by a red circle in the center of the image). The right picture shows the mounting of the same sensor in detail.

Appropriate and exact labels for the acceleration data are necessary for training and evaluation. These are read out synchronously from the programmable logic controller of the injection molding machine. In total, there are five different states representing the current process step of the machine that can occur: closing, injection unit, plasticizing, demolding and null state—where the null state describes the state in which none of the other four states is active.

The experiments presented in this paper used production data from the manufacture of 15 different products recorded on five different injection molding machines. All machines are from the Victory series of the Austrian machine manufacturer Engel. The years of construction of the five machines used range from the mid-1990s to the present.

We can see in Figure 2 that the distribution and structure of states varies greatly between different data sets. In particular, the difference in the realizations can be seen most clearly in the plasticizing state. In (a) and (b), the energy in the signal is significantly lower than in (c).

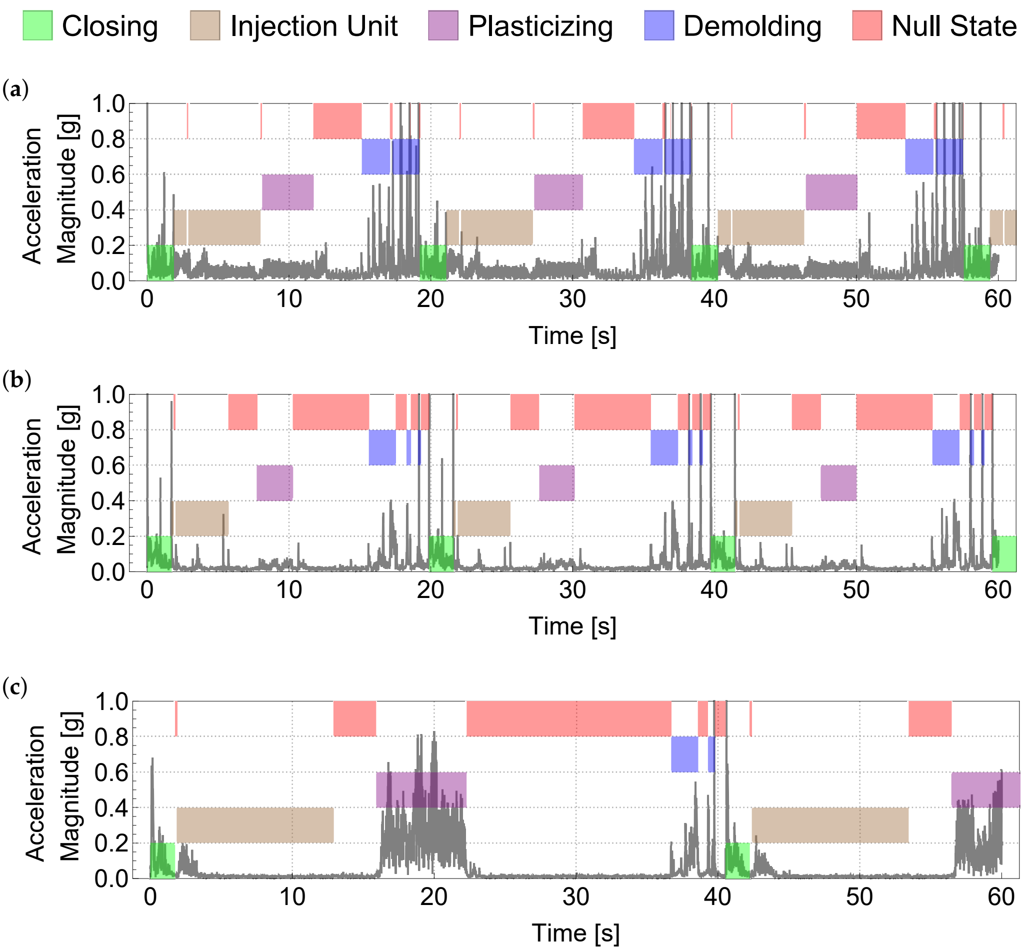

Figure 2. Depiction of excerpts from three different datasets (**a**–**c**). The injection molding cycles shown in (**a**,**b**) are rather short cycles of slightly more than 20 s, (**c**) shows a slightly longer cycle of about 40 s in length.

One interesting point is to use acceleration data to identify when the machine is producing and when it is not. For this purpose, we introduce the terms producing and non-producing. Producing means that the states closing, injection unit, plasticizing, demolding and closing occur one after the other. This sequence of states suggests that a plastic part has been produced, and an injection molding cycle has been completed—a different sequence of states is evaluated as non-producing. In addition, if the null state is active for too long between two cycles, this is considered non-producing.

4. Data Analysis

4.1. Pre-Processing

Besides the vibration of the machine, the sensor also detects a constant acceleration due to gravity. To eliminate this effect, the first step is to subtract a running mean from each axis. Afterwards, the magnitude of the acceleration is calculated:

$$M(i,n) = \sqrt{\left(x_i - \sum_{j=-n}^{n} \frac{x_{i+j}}{2n+1}\right)^2 + \left(y_i - \sum_{j=-n}^{n} \frac{y_{i+j}}{2n+1}\right)^2 + \left(z_i - \sum_{j=-n}^{n} \frac{z_{i+j}}{2n+1}\right)^2}, \quad (1)$$

for $i = n+1, n+2, \ldots$, where (x_i, y_i, z_i) denotes the respective values of time series at time i. An example of the original and resulting time series for $n = 100$ is depicted in Figure 3 below:

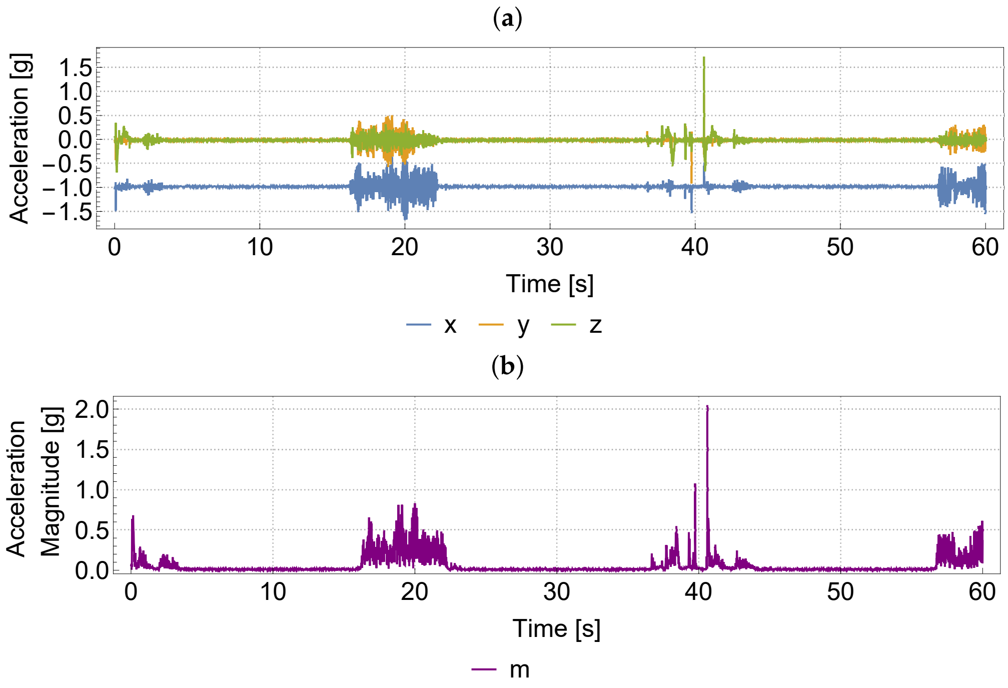

Figure 3. Depiction of the pre-processing step: (**a**) original 3D data set; (**b**) transformed 1D time series acceleration magnitude $M(\cdot, 100)$.

The resulting time series with elements (1) has more contrasting segments characterised by low and high variances which is clearly an advantage for further classification tasks.

4.2. Tree Based Approach

When selecting possible methods for analyzing the data, it was necessary to use automatable machine learning methods. The reason for this is that it is assumed that the underlying data lake will become larger and larger. It is assumed that, in the future, it will be necessary to repeatedly train new models over an ever larger data set. For such adaptations to be possible in an efficient manner, it is necessary to rely on machine learning. Within the machine learning methods, two established methods, namely a tree-based approach and an artificial neural network, were chosen because they are established and well applicable methods.

For our first approach, we extract 44 statistical features from the pre-processed magnitude of acceleration. Twenty-two of these features stem from the original time-series, and 22 from the first order differences of our signals, which may be interpreted as an estimator for the jerk or jolt, which is defined as the first derivative of the acceleration. To this end, we take a running window of length 200 ms for which we calculate said features. The features are then labeled with the class label positioned at the 100th time-step in such an interval. The resulting feature set is then under-sampled, which is achieved by randomly sampling from each class to attain equal class frequencies. Finally, we apply a gradient boosted tree algorithm on this reduced set which we then use to predict the labels of the yet unseen current test set.

4.3. Artificial Neural Network

In our neural network approach, we were influenced by the idea of semantic segmentation, a well-known computer vision problem in which each pixel of an image is assigned a label. In the context of time series data, this means that we predict a label for every timestamp in the input series. To achieve this, we implement a 1D fully-convolutional network, this allows us to get a prediction for each input value. As the time duration of the states can vary greatly, we decide to counteract this problem by applying convolutions with various receptive fields to the same input time series. This idea is based on the work presented in [25]. The key difference is that we adapt the architecture in such a way that it predicts a label for every timestamp in the input series. This is a significant change that allows 1D convolutions to be used for segmenting time series data. In Figure 4, a sketch of one such convolutional block is shown.

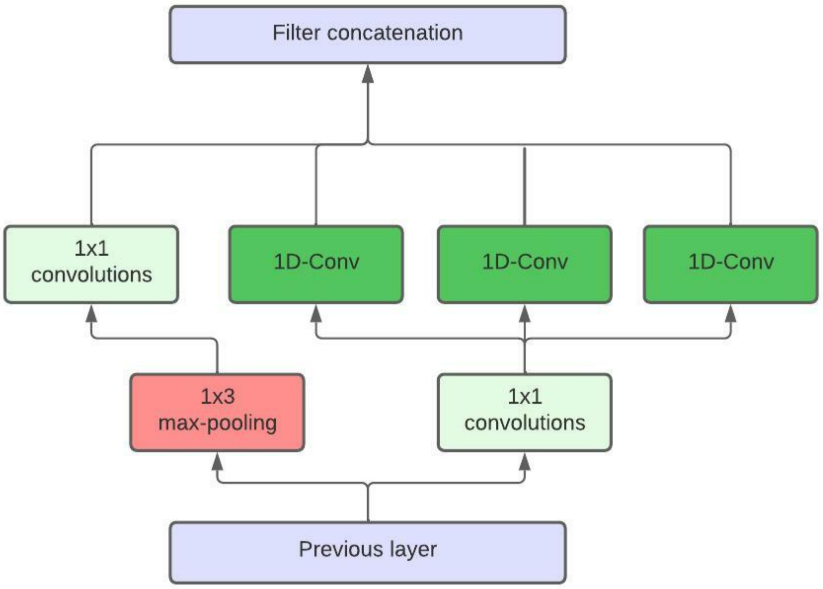

Figure 4. Our Inception module for the time series segmentation. Note that the three 1D-Conv blocks marked in green have different receptive fields.

The neural network model can be interpreted as a function, $g : \mathbb{R}^{MxT} \to \mathbb{R}^{KxT}$, where T indicates the length of the input time series, K the number of different classes and M represents the number of input values per time stamp. In our case, $M = 4$ consists of the acceleration data of the three axes as well as their magnitude. Since we are dealing with a multi-class problem, the cross entropy loss function is used as objective function for the training. For the setting in this paper, the cross entropy error for one prediction is depicted in equation (2), where $\hat{\mathbf{Y}}_n \in \mathbb{R}^{K\times T}$ is the model prediction and $\mathbf{Y} \in \mathbb{R}^{K\times T}$ the corresponding label:

$$CE(\mathbf{Y}, \hat{\mathbf{Y}}) = -\sum_{t=1}^{T}\sum_{c=1}^{K} y_{c,t} \cdot log(\hat{y}_{c,t}) \text{ , with } \hat{y}_{c,t} \in [0,1] \text{ and } y_{c,t} \in \{0,1\}. \tag{2}$$

In order to minimize the cross entropy loss, we use the *Adam* [26] optimizer with a learning rate of $2e - 4$.

5. Results and Numerical Analysis

This section describes the two experiments carried out in this work. In the first subsection, the used metrics are described. After that the two different experimental settings, including their results, are presented. For the considered states which are described in Section 3, we use the following abbreviations:

1 → Closing, 2 → Injection Unit, 3 → Plasticizing, 4 → Demolding, 5 → Null State

5.1. Metrics

The choice of the right performance measurement is a crucial task in a machine learning project. For classification problems, the accuracy is an often used metric because of its simplicity and interpretability. Besides the accuracy, there are plenty of other metrics which can be used to evaluate and analyse classification results. In [27], the most common performance measures are described and compared. To assess the classification quality, we resort to confusion matrices and different performance metrics. A confusion matrix [28] C of an algorithm A on some dataset $\mathcal{D}$ consists of entries $C_{i,j}$, which are defined as "Number of examples of class i in $\mathcal{D}$ classified as j by A". As our matrices contain 25 entries each, we also state multiple performance metrics, which can be interpreted as a function of the entries of a confusion matrix and concisely represent the classification quality. We adjust the confusion matrices presented is this article such that they are rescaled for every true state, i.e., every row adds up to 100 (percent). Since we are confronted with an imbalanced class-distribution, we decided that, for balanced accuracy (BACC), as defined in (3) below, as an internal performance metric during the training of classical algorithm:

$$BACC(C) = \frac{1}{5}\sum_{i=1}^{5} \frac{C_{i,i}}{\sum_{j=1}^{5} C_{i,j}} \tag{3}$$

$$ACC(C) = \frac{\sum_{i=1}^{5} C_{i,i}}{\sum_{i=1}^{5}\sum_{j=1}^{5} C_{i,j}} \tag{4}$$

$$F\text{-}Score(C) = \frac{1}{5}\sum_{i=1}^{5} \frac{2C_{i,i}}{\sum_{j=1}^{5} C_{i,j} + \sum_{j=1}^{5} C_{j,i}} \tag{5}$$

Since we employed a cross-validation scheme, we add all confusion matrices for a final resulting matrix, for which we calculated and present the results in the following section.

5.2. Numerical Analysis

In the present work, we employ two evaluation methods for our proposed systems. The first one, called Inter-tool validation, always splits the data in such a way that all data from one tool is either in the train dataset or in the test data set. For the second experiment, called Intra-tool validation, the database is randomly split into a train dataset and into a test data set. With these evaluation methods, we are able to analyse if our methods learn injection moulding process specific characteristics or if only tool dependent ones are learned.

5.2.1. Experiment I: Inter-Tool Validation Results

The available database consists of data from 15 different injection tools. In this experiment, a five-fold cross-validation is performed in the manner that all data from exactly three tools are used as the test data and the remaining data for training. Below, the summed up confusion matrices, as well as the averaged metrics, are presented.

The inspection of our two confusion matrices depicted in Tables 1 and 2 implies that both algorithms tend to confuse state 2 (Injection Unit) and state 5 (Null State). This may be attributed to the general nature of our Null State and the rather brief occurrences of injection.

Table 1. Experiment I: Confusion matrix of our tree-based algorithm.

		Predicted Labels				
		1	2	3	4	5
True Labels	1	84.6	2.60	0.106	12.4	0.229
	2	8.13	53.9	8.49	0.518	28.9
	3	0.217	14.4	79.6	4.37	1.40
	4	15.8	0.368	0.0124	81.0	2.77
	5	2.64	18.9	3.56	10.6	64.2

Table 2. Experiment I: Confusion matrix of our neural network.

		Predicted Labels				
		1	2	3	4	5
True Labels	1	84.8	1.95	3.82	7.71	1.68
	2	1.27	64.4	8.92	1.20	24.2
	3	10.7	8.60	78.5	0.965	1.18
	4	7.55	1.21	0.353	88.4	2.52
	5	2.30	35.2	4.42	3.65	54.4

In addition to these evaluations, we also conducted a statistical evaluation to assess whether there is a statistically significant difference between our two chosen approaches. To this end, we depicted three performance measures as box-whisker charts and compared the values of all 15 datasets using a classic one-way ANOVA with repeated measures.

From Figures 5–7, the similar performance of our two methods when considering our three performance measures is apparent. In Table 3, we present results from conducting a two-sided test for equal means, which corroborates with the impression of similar performance. What we can infer from Figures 5–7 is that the highest and lowest values of each performance measure are attained by the neural network approach.

Table 3. Summary of results from our Experiment I.

Model	**BACC**	**ACC**	**F-Score**
Classic	0.7268	0.6846	0.6829
NN	0.7411	0.6698	0.7002

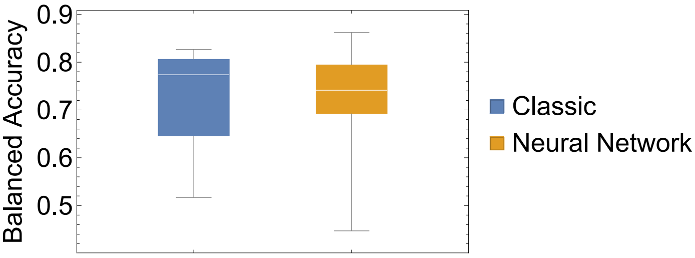

Figure 5. Experiment I: Box whisker chart of BACC values of our 15 datasets for both considered methods.

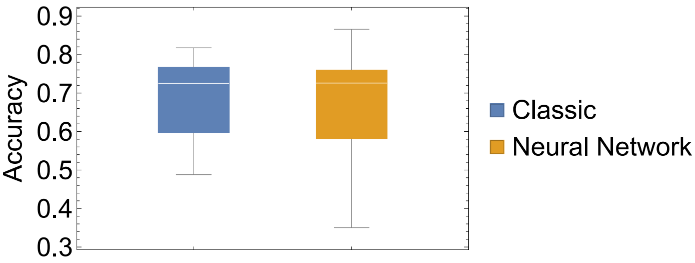

Figure 6. Experiment I: Box whisker chart of ACC values of our 15 datasets for both considered methods.

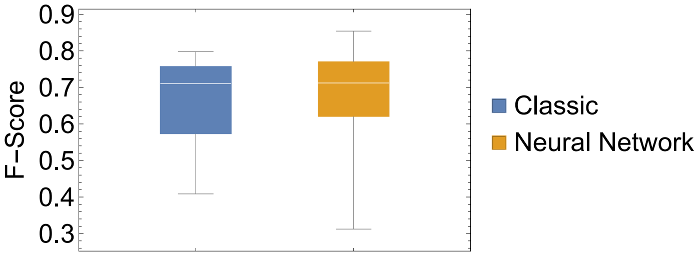

Figure 7. Experiment I: Box whisker chart of F-Score values of our 15 datasets for both considered methods.

The high p-values presented in Table 4 do not indicate any significant difference in all performance measures.

Table 4. Statistical comparison of our results based on performance measures for each data-set.

Perf Measure	Balanced Acc.	Accuracy	F-Score
***p*-value**	0.9871	0.7837	0.9281

5.2.2. Experiment II: Intra-Tool Validation Results

Our second experiment considers a different data split up, such that we have each machine–tool combination already in the data set. To this end, we split up the data into training and test data, where the test data consist of 20% of each of our 15 data sets. This split up is repeated five times, and both algorithms are learned on each training set and applied on each test set, respectively, as described in in the introduction to section 5.2.

From inspecting the two confusion matrices in Tables 5 and 6 and the performance measures in Table 7, it is evident that including data from a machine on which the algorithms are tested into the training set significantly increases the classification quality. When comparing the results from our tree classifier with the performance in Experiment I, we can see that the classification error (1-ACC) reduces from 0.3154 to 0.1973, a reduction of about 37.45%. The same comparison for our NN approach yields an error reduction from 0.3302 to 0.0834, a decrease of about 74.74%. Moreover we can observe that, while, for the results from Experiment I, the performance values are rather similar, in our Experiment II, the respective values differ greatly. The Neural Network attains an average BACC value of 0.9413, while the tree algorithm achieves 0.8571. In terms of error, this translates to an error that is 58.92% lower for our neural network. In case of ACC, we also observe an error reduction of more than 50%. These observations suggest that the generalisation power of our algorithms to unseen machine/tool combinations still has optimization potential. We assume that a larger and even more diverse data-base would act as a partial remedy.

Table 5. Experiment II: Confusion matrix of our tree-based algorithm.

		Predicted Labels				
		1	2	3	4	5
True Labels	1	94.7	1.34	0.651	3.83	0.0385
	2	3.39	72.1	1.50	0.0316	23.0
	3	0.0524	2.96	96.0	0.123	0.854
	4	3.50	0.221	0.	94.6	1.65
	5	2.18	12.5	4.21	10.0	71.1

Table 6. Experiment II: Confusion matrix of our neural network.

		Predicted Labels				
		1	2	3	4	5
True Labels	1	98.8	1845	524	1053	1415
	2	0.231	85.7	0.412	0.121	13.5
	3	0.0705	0.337	99.1	0.0303	0.492
	4	0.187	0.133	0.0286	98.9	0.714
	5	0.321	10.1	0.755	0.761	88.1

Table 7. Summary of results from our Experiment II.

Model	BACC	ACC	F-Score
Classic	0.8571	0.8027	0.8152
NN	0.9413	0.9166	0.9338

6. Conclusions

This paper has proposed a monitoring system for classifying five different states in the injection molding process. We evaluated two different machine learning approaches, for which we recorded acceleration data using a sensor attached to the injection molding machine. The corresponding labels for the model training were obtained from the programmable logic controller. Both methods were evaluated in two different experiment settings. In the first experiment, the training and test data differ in the way that the test dataset contains injection moulding cycles of tools that were not included in the training data. For this setting, both methods perform quite similarly with a balanced accuracy of 72% and 74%. The main difference is that the results from the neural network approach show higher variance. This is due to the fact that the acceleration data have very different characteristics regarding the used tools the sensor was mounted on. In the second experiment, the collected datasets from all tools were combined and then randomly split into a train and test set. Here, the performance of both models increases up to 86% and 94%.

These results show that there is a high potential in detecting various states in the injection moulding cycle. Nevertheless, the results also suggest that the proposed methods still show a lack of generalization. In order to improve this weakness we suggest using a larger dataset with a higher diversity of injection tools.

Author Contributions: Conceptualization, V.S., W.L. and D.E.; methodology, J.B., P.G., V.S., W.L. and D.E.; software, P.G., J.B., W.L. and V.S.; validation, P.G., J.B., W.L. and V.S.; resources and data curation, P.G. and J.B.; writing—original draft preparation, V.S., J.B. and P.G.; writing—review and editing, J.B., P.G., V.S., W.L. and D.E.; visualization, V.S.; supervision, W.L. and D.E.; project administration, W.L.; funding acquisition, W.L., V.S. and D.E. All authors have read and agreed to the published version of the manuscript.

Funding: This work has been supported by the COMET-K2 Center of the Linz Center of Mechatronics (LCM) funded by the Austrian federal government and the federal state of Upper Austria. AISEMO's contribution was supported by the Austrian Research Promotion Agency (FFG project number FO999890614).

Informed Consent Statement: Not applicable

Data Availability Statement: The authors can be contacted to obtain data used in the study.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results

References

1. Struchtrup, A.S.; Kvaktun, D.; Schiffers, R. Comparison of feature selection methods for machine learning based injection molding quality prediction. *AIP Conf. Proc.* **2020**, *2289*, 020052.
2. Obregon, J.; Hong, J.; Jung, J.Y. Rule-based explanations based on ensemble machine learning for detecting sink mark defects in the injection moulding process. *J. Manuf. Syst.* **2021**, *60*, 392–405. [CrossRef]
3. Polenta, A.; Tomassini, S.; Falcionelli, N.; Contardo, P.; Dragoni, A.F.; Sernani, P. A Comparison of Machine Learning Techniques for the Quality Classification of Molded Products. *Information* **2022**, *6*, 272. [CrossRef]
4. Baturynska, I.; Martinsen, K. Prediction of geometry deviations in additive manufactured parts: Comparison of linear regression with machine learning algorithms. *J. Intell. Manuf.*,**2021**, *1*, 179–200. [CrossRef]
5. Tercan, H.; Guajardo, A.;Heinisch, J.; Thiele, T.; Hopmann, C.;Meisen, T. Transfer-learning: Bridging the gap between real and simulation data for machine learning in injection molding. *Procedia CIRP* **2018**, *72*, 185–190. [CrossRef]
6. Schmid, M.; Altmann, D.; Steinbichler; G. A Simulation-data-based machine learning model for predicting basic parameter settings of the plasticizing process in injection molding. *Polymers* **2021**, *13*, 2652. [CrossRef]
7. Hürkamp, A.; Gellrich, S.; Ossowski, T.; Beuscher, J.; Thiede, S.; Herrmann, C.;Dröder, K.. Combining simulation and machine learning as digital twin for the manufacturing of overmolded thermoplastic composites. *J. Manuf. Mater. Process.* **2020**, *3*, 92. [CrossRef]
8. Chen, J.C.; Guo, G.; Wang, W.N. Artificial neural network-based online defect detection system with in-mold temperature and pressure sensors for high precision injection molding. *Int. J. Adv. Manuf. Technol.* **2020**, *7*, 2023–2033. [CrossRef]
9. Gordon, G.;Kazmer, D.O.; Tang, X.; Fan, Z.; Gao, R.X. Quality control using a multivariate injection molding sensor. The *Int. J. Adv. Manuf. Technol.* **2015**, *9*, 1381–1391. [CrossRef]
10. Lee, H.; Liau, Y.; Ryu, K. Real-time parameter optimization based on neural network for smart injection molding. *IOP Conf. Ser. Mater. Sci. Eng.* **2018**, *324*, 012076. [CrossRef]
11. Tsai, M.H.; Fan-Jiang, J.C.; Liou, G.Y.; Cheng, F.J.; Hwang, J.; Peng, H.S.; Chu, H.Y. Development of an online quality control system for injection molding process. *Polymers* **2022**, *8*, 1607. [CrossRef] [PubMed]
12. Chen, J.Y.; Tseng, C.C.; Huang, M, Quality indexes design for online monitoring polymer injection molding. *Adv. Polym. Technol.* **2019**, *2019*, 3720127. [CrossRef]
13. Zhang, J.; Zhao, P.; Zhao, Y.; Huang, J.; Xia, N.; Fu, J. On-line measurement of cavity pressure during injection molding via ultrasonic investigation of tie bar. *Sensors Actuators Phys.* **2019**, *285*, 118–126. [CrossRef]
14. Ageyeva, T.; Horváth, S.; Kovács, J.G. In-mold sensors for injection molding: On the way to industry 4.0. *Sensors* **2019**, *16*, 3551. [CrossRef] [PubMed]
15. Selvaraj, S.K.; Raj, A.; Mahadevan, R.R.; Chadha, U.; Paramasivam, V. A Review on Machine Learning Models in Injection Molding Machines. *Adv. Mater. Sci. Eng.* **2022**, *2022*, 1949061. [CrossRef]
16. Werner, M.; Schneider, M.; Greifzu, N.; Seul, T.; Wenzel, A.; Jahn, R. Development of a digital assistance system for continuous quality assurance in the plastic injection molding process with a focus on self-learning algorithms. *SPE-Inspiring Plast. Prof.* **2020**, 21–25.
17. Charest, M.; Finn, R.; Dubay, R. Integration of artificial intelligence in an injection molding process for online process parameter adjustment. In Proceedings of the 2018 Annual IEEE International Systems Conference (SysCon), Vancouver, BC, Canada, 23–26 April 2018; pp. 1–6.
18. Jung, H.; Jeon, J.; Choi, D.; Park, J.-Y. Application of Machine Learning Techniques in Injection Molding Quality Prediction: Implications on Sustainable Manufacturing Industry. *Sustainability* **2021**, *13*, 4120. [CrossRef]
19. Nagorny, P.; Pillet, M.; Pairel, E.; Goff, R.L.; Loureaux, J.; Wali, M.; Kiener, P. Quality prediction in injection molding. In Proceedings of the 2017 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA), Annecy, France, 26–28 June 2017; pp. 141–146.
20. Ketonen, V.; Blech, J.O. Anomaly Detection for Injection Molding Using Probabilistic Deep Learning. In Proceedings of the 2021 4th IEEE International Conference on Industrial Cyber-Physical Systems (ICPS), Victoria, BC, Canada, 10–12 May 2021; pp. 70–77.
21. Ke, K.-C.; Huang, M.-S. Quality Classification of Injection-Molded Components by Using Quality Indices, Grading, and Machine Learning. *Polymers* **2021**, *13*, 353. [CrossRef]
22. Fan, S.; Jia, Y.; Jia, C. A Feature Selection and Classification Method for Activity Recognition Based on an Inertial Sensing Unit. *Information* **2019**, *10*, 290. [CrossRef]
23. Moreira, E.E.; Alves, F.S.; Martins, M.; Ribeiro, G.; Pina, A.; Aguiam, D.E.; Sotgiu, E.; Fernandes, E.P.; Gaspar, J. Industry 4.0: Real-time monitoring of an injection molding tool for smart predictive maintenance. In Proceedings of the 2020 25th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), Vienna, Austria, 8–11 September 2020; pp. 1209–1212.
24. Zhang, J.Z. Development of an in-process Pokayoke system utilizing accelerometer and logistic regression modeling for monitoring injection molding flash. *Int. J. Adv. Manuf. Technol.* **2014**, *71*, 1793–1800. [CrossRef]

25. Hassan, I.F.; Lucas, B.; Forestier, G.; Pelletier, C.; Schmidt, D.F.; Weber, J.; Webb, G.I.; Idoumghar, L.; Muller, P.-A.; Petitjean, F. InceptionTime: Finding AlexNet for time series classification. *Data Min. Knowl. Discov.* **2020**, *34*, 1936–1962.
26. Kingma, D.P.; Ba, J. Adam: A Method for Stochastic Optimization. In Proceedings of the International Conference on Learning Representations, San Diego, CA, USA, 7–9 May 2015.
27. Ferri, C.; Hernández-Orallo, J.; Modroiu, R. An experimental comparison of performance measures for classification. *Pattern Recognit. Lett.* **2009**, *30*, 27–38. [CrossRef]
28. Ballabio, D.; Grisoni, F.; Todeschini, R. Multivariate comparison of classification performance measures. *Chemom. Intell. Lab. Syst.* **2018**, *174*, 33–44. [CrossRef]