The 9th European Congress on Computational Methods in Applied Sciences and Engineering
ECCOMAS Congress 2024
3–7 June 2024, Lisboa, Portugal

# ENHANCING INDUSTRIAL QUALITY CONTROL EFFICIENCY: AN INNOVATIVE DEEP LEARNING APPROACH FOR SUSTAINABLE PROCESS MONITORING

**MERIAM ZRIBI[1,∗,†], PAOLO PAGLIUCA[2,†] AND FRANCESCA PITOLLI[1]**

[1] Department of Basic and Applied Sciences for Engineering
Università degli Studi Roma La Sapienza
Via Antonio Scarpa 14, Roma
e-mail: {meriam.zribi, francesca.pitolli}@uniroma1.it

[2] Institute of Cognitive Sciences and Technologies
National Research Council (CNR)
Via Gian Domenico Romagnosi 18/A, Roma
email: paolo.pagliuca@istc.cnr.it

**Key words:** Sustainability, environmental concern, process monitoring, deep neural networks

**Summary.** *A major concern in traditional industrial monitoring is the strong environmental impact, mainly related to inefficiency of classic paradigms. In fact, in many sectors, monitoring systems typically rely on the presence of human operators responsible for the detection of errors or faults. However, this activity is heavily influenced by many factors like subjectivity or physical conditions (e.g., fatigue, lighting), making this strategy ineffective in terms of costs (both environmental and company-wide) and results. For instance, when the process involves the control of production lots, if the operator identifies any anomalies the whole batch might be discarded. Sustainability and performance can be achieved by the automation of the monitoring process. In this regard, we propose an innovative method based on a deep neural network that can discriminate between correct and faulty items in a production batch. Our model allows to significantly reduce disposal costs, since it analyzes each item rather than considering the whole batch, thus preventing the waste of potentially usable resources. Furthermore, the methodology enables the optimization of the monitoring quality and lightens the responsibilities of the human operator, who only reviews the model outputs and generates relevant statistics for the company. We provide a thorough description of the proposed model in the context of the monitoring of transparent tubes within the production process of a company dealing with plastic consumables. Preliminary experiments we have performed on a real dataset confirm the effectiveness of the proposed method.*

∗ Corresponding author
† These authors contributed equally to this work.

## 1 INTRODUCTION

The impact of human activity on environment over the past decades is undeniable [1, 2]: industrialization is responsible for the deterioration of the environmental conditions due to a significant increase of the gas emissions, water pollution and usage of toxic materials [3–5]. In particular, classic industrial paradigms have a strong impact on the amount of waste of resources and energy consumption. For instance, industrial process monitoring very often requires the presence of human operators, whose activity is influenced by several factors like subjectivity, physical conditions (e.g., stress and/or fatigue), lighting and others. In the case of the control of a production lot, when the operator detects a fault, the entire lot might be discarded independently of the actual amount of defective items. This strategy is not effective and has a strong impact on the environment (the disposal costs are not negligible). Instead, modern monitoring processes should aim at improving the quality of processes in order to optimize performance, reduce costs and preserve a sustainable environment, which has been defined as "meeting the needs of the present without compromising the ability of future generations to meet their own needs." [6]. There have been several attempts in which mechanisms to increase sustainability in the control process have been introduced [7–9]. In this respect, the Industry 4.0 paradigm [10] represents an opportunity to change the way of thinking industrial processes [11–14] through the exploitation of Artificial Intelligence (AI) [15]. In fact, AI enables to collect a large amount of data and use advanced techniques in order to increase sustainability, efficiency, flexibility and robustness as well as reduce costs and waste of materials. In [16] the authors employ the YOLOv4 model [17] to the disposable gas lighter manufacturing process. They prove that the approach is sustainable in terms of accuracy and processing time and applicable to a real manufacturing process. In [18] a deep learning-based computer vision approach is proposed, which tackles the issue of checking the wear state of products. In particular, the authors provide evidence of performance and reduced emission of $CO_2$ (i.e., sustainability).

To increase the sustainability and efficiency of process monitoring, we propose an automated model able to detect faulty items. Specifically, we focus on the monitoring of transparent tubes within the production process of a company coping with plastic consumables. We used a deep Convolutional Neural Network (CNN) for the ability to identify the presence or the absence of an anticoagulant inside the tubes. In particular, we employed the ConvNet3_4 model [19, 20] with the addition of pooling layers [21] in the convolutional part of the model. As stated by the authors: "In all cases, pooling helps to make the representation become approximately invariant to small translations of the input. Invariance to translation means that if we translate the input by a small amount, the values of most of the pooled outputs do not change." [21]. The addition of pooling allows to reduce the consumption of resources like time, memory or energy while preserving excellent performance, thus enhancing sustainability. The results obtained on a preliminary dataset demonstrate the validity of the approach, with a perfect capability of detecting faulty tests. Moreover, the proposed model is computationally more efficient than the original ConvNet3_4 model. This is a feature that turns out to be of critical importance in real-world scenarios, where minimizing the computational load has considerable effects on sustainability, environmental concerns, disposal costs, performance and scalability. In particular, the analysis of consumed resources (i.e., memory, CPU, processing time) reveals how the lightweight version of the ConvNet3_4 model introduced in this work represents the best solution to guarantee sustainability and efficiency both during a post-evaluation phase and in a real acquisition scenario.

In the next sections we present the methodology we used (Section 2) and the obtained outcomes (Section 3). Our conclusions are reported in Section 4.

## 2 MATERIALS AND METHODS

As we pointed out in Section 1, the aim of this work is to automate the monitoring of transparent tubes, aiming at enhancing sustainability and efficiency of the whole process. Specifically, we used the ConvNet3_4 model [19, 20] with the addition of pooling layers in the convolutional part of the network (i.e., there is one pooling layer after each convolutional layer). We refer to it as the "ConvNet3_4+pool" model. For a detailed description of the original ConvNet3_4 model, the reader is referred to [20]. Concerning the pooling operation, it is generally applied to each convolutional layer aiming to reduce the spatial size of the convoluted features and extract the most relevant one: given an input image $I$ of size $m_1 \times m_2 \times m_c$ (where $m_1$ denotes the height of the image, $m_2$ indicates the width of the image and $m_c$ represents the number of channels of the image), the output of a generic convolutional layer $l$ (with $l > 2$) is provided in Eq. 1:

$$F_l = Conv(F_{l-1}, K) \qquad l = 2, ..., N_{conv\_layers} \tag{1}$$

where $K$ is a kernel of size size $n_1 \times n_2 \times n_c$, $F_{l-1}$ denotes the output of previous convolutional layer ($F_1 = I$), $N_{conv\_layers}$ represents the number of convolutional layers and $Conv()$ indicates the convolution operation (see [20], page 6). The pooling operation is applied to the convolutional layer $F_l$ as indicated in Eq. 2:

$$P_l = \phi_p(F_l) \qquad l = 1, ..., N_{conv\_layers} \tag{2}$$

where $\phi_p()$ represents the pooling function (in this work we use the *MaxPooling* function) and $P_l$ is the output of pooling layer. The size of $P_l$ is $\frac{(m_1+2\times p-n_1)}{s} \times \frac{(m_2+2\times p-n_2)}{s} \times m_c$, with $s$ indicating the stride (see [20], Table 1) and $p$ denoting the padding, which is used to add extra pixels around the boundary in order to increase the size of the output of the pooling layer.

The ConvNet3_4+pool model has to categorize tubes according to the presence or absence of an anticoagulant inside them. The model receives as inputs images of size $400 \times 400$ pixels like those shown in [19], Figure 1. The original dataset contained 402 images, equally split into the two classes (i.e., present and absent). However, because of the limited size, we created a larger dataset through data augmentation technique [22, 23]. The usage of a high number of input data enables to consistently reduce the overfitting issue [22]. The set of transformations we used are described in Table 1. We employed only those transformations leading to realistic images. Overall, the dataset used to train the ConvNet3_4+pool model contains 5531 images. The training set is made of 4433 images (around 80% of the dataset), while the test set is constituted by 1098 images (around 20% of the dataset).

In order to reduce the risk of discovering a good model only due to lucky initialization, we repeated the training of the ConvNet3_4+pool model 10 times, each one using a different network initialization. The parameter settings are shown in Table 2 and are derived from [20]. Concerning the size of the pooling layers, we set it to 3.

## 3 RESULTS

This section contains a detailed description of the outcomes we achieved. We adopted the Mann-Whitney U test for statistical analyses.

In order to assess whether our model is suitable for the given problem and avoid overfitting on the input images, we employ the widely used k-fold cross-validation technique [25–27], with

Table 1: Set of transformations used to widen the size of the input data. Regarding the RandomRotation transformation [24], we slightly modified the original rotation in order to ensure that the amount of rotation corresponds to the value indicated in the parameter $angle$. With respect to the RandomAdjustSharpness transformation [24], the parameter $sharpness\_factor$ determines how much the sharpness of the image is increased (values higher than 1) or reduced (values lower than 1). The case $sharpness\_factor = 1$ has not been considered, since it corresponds to cloning the input image. The parameter $prob$ has been set to 1 in order to guarantee the generation of a different image. The parameter $size$ refers to CenterCrop transformation [24] and is expressed in terms of $[Height, Width]$ (i.e., the output is an image of size $Width \times Height$ pixels), while the parameters $padding$ and $fill$ concern the Pad transformation [24] and are used respectively to restore the size of the image to $400 \times 400$ pixels and specify their color. The parameter $fill$ has been set to 0 in order to avoid the introduction of meaningless data that might alter the model's prediction. For further details, see [24].

| Transformation | Parameters |
|---|---|
| RandomRotation | $angle \in [1°, 2°, 3°, 4°, 5°, 6°, 7°, 8°, 9°, 10°]$ |
| RandomAdjustSharpness | $sharpness\_factor \in [0, 0.5, 1.5, 2], prob = 1.0$ |
| CenterCrop + Pad | $size \in [(390, 390), (380, 380), (360, 360)]$<br>$pad \in [5, 10, 20], fill = 0$ |

Table 2: Training parameters. We kept the same settings as in [20]. With respect to the pooling layers, the size has been set to 3.

| Parameter | Value |
|---|---|
| # of replications | 10 |
| # of epochs | 50 |
| batch size | 64 |
| learning rate | $10^{-4}$ |
| weight decay | $10^{-2}$ |
| pooling size | 3 |

$k = 5$. We trained both the ConvNet3_4 and the ConvNet3_4+pool models on the dataset obtained by applying the transformations described in Table 1 and we obtained a perfect 100% accuracy in either cases (data not shown). However, we observed remarkable differences in terms of sustainability and environmental costs. If we analyze the resources used by the two models, it is evident that the ConvNet3_4+pool is more sustainable (see Table 3): it requires a significant less amount of time for training, which corresponds to a noticeable efficiency in terms of energy consumption (i.e., CPU, memory). Furthermore, the size of the model is of almost three orders of magnitude smaller than that of the ConvNet3_4 model. The latter has potential applications in real scenarios: using a lightweight model is fundamental when dealing with limited hardware resources.

Table 3: Required resources: (a) time necessary to train the model in hours and (b) size of the model in megabytes (MB). Best data are denoted in bold (lower values correspond to better efficiency).

| Model | Time | Size |
|---|---|---|
| ConvNet3_4 | 6 | 931.5 |
| ConvNet3_4+pool | **1.1** | **1.1** |

Table 4: Set of transformations used to create the test set: the parameters used represent tiny variations of those reported in Table 1.

| Transformation | Parameters |
|---|---|
| RandomRotation | $angle \in [0.5°, 1.5°, 2.5°, 3.5°, 4.5°]$ |
| RandomAdjustSharpness | $sharpness\_factor \in [2.5, 3], prob = 1.0$ |
| CenterCrop + Pad | $size = (320, 320)$<br>$pad = 40, fill = 0$ |

Aiming to investigate the energy consumption of the different models, we carried out a post-evaluation phase by using a test set built by means of the transformations shown in Table 4. This set contains 400 images equally split into the two categories (i.e., "present" and "absent"). We measured the memory and CPU usage during the evaluation of the ConvNet3_4 and ConvNet3_4+pool models, as well as the processing time. Data are illustrated in Fig. 1. As it can be seen, the latter model uses around 3 times less memory than the former throughout the post-evaluation phase (Fig. 1, a). Furthermore, the processing time of the ConvNet3_4+pool model is approximately 10 times lower than that of the ConvNet3_4 model (Fig. 1, c). Concerning the CPU usage, both models display similar average consumption (49.986% versus 49.919%, see Fig. 1, b and Table 5). Overall, these outcomes clearly demonstrate that the ConvNet3_4+pool model enables to consistently reduce the amount of computational required resources, hence enhancing a higher sustainability.

The outcomes presented so far are limited to the analysis of the ConvNet3_4 model, which

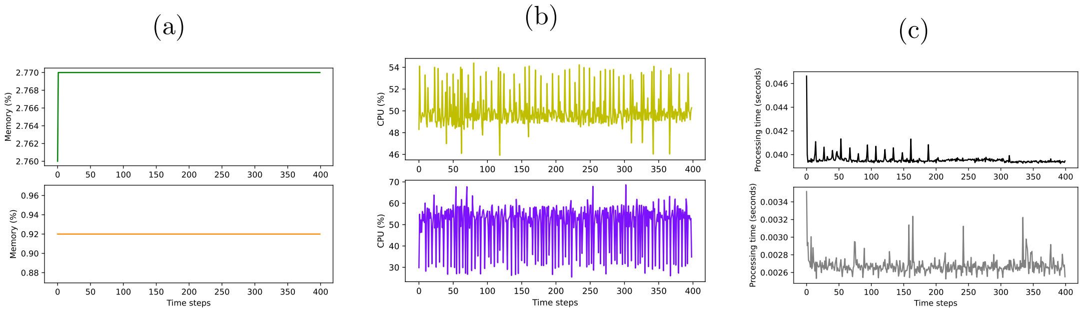


Figure 1: Resources used by the ConvNet3_4 (top) and the ConvNet3_4+pool (bottom) models. We collected data by measuring the consumption of resources during the entire post-evaluation phase. Specifically, we computed the amount of memory (a), CPU (b) and processing time (c) required by the models to classify the 400 images in the test set. Measures have been calculated at each time step, which corresponds to the time required by the model to perform the classification of an input image.

proved successful in similar settings [19, 20], and its lightweight version (ConvNet3_4+pool) introduced in this work. Nonetheless, one might argue that using simpler networks, consisting of a smaller number of layers or characterized by fewer parameters, could be more beneficial for sustainability, since the amount of memory and CPU might be considerably reduced. To this end, we compared the two aforementioned models with other models we built from scratch:

- ConvNet1_2: it has one convolutional layer and two fully-connected layers;
- ConvNet1_2+pool: it corresponds to the version of ConvNet1_2 model with the addition of a pooling layer of size 3;
- ConvNet2_3: it consists of two convolutional layer and three fully-connected layers;
- ConvNet2_3+pool: it represents the version of ConvNet2_3 model with the addition of pooling layers of size 3 in the convolutional part of the model.

The aim of this analysis is to demonstrate that the ConvNet3_4 and ConvNet3_4+pool models are better in terms of accuracy without any significant impact on the amount of consumed resources. The obtained results indicate similar performance in terms of sustainability (i.e., CPU and memory usage) among all the models exploiting pooling layers (see Table 5), while the ConvNet3_4 model displays a slightly higher memory and CPU consumption than the ConvNet1_2 and ConvNet2_3 models (see Table 5). It was interesting to notice that among the models with pooling operations, the one that turned out to be the most resource-intensive was actually the simplest, characterized by fewer layers. Therefore, it's not necessarily the case that a simpler model is more sustainable, even though this was observed, for example, among the models without pooling operations. In addition, we collected data about accuracy of all the presented models (see Fig. 2). As it can be observed, the ConvNet3_4 and ConvNet3_4+pool are the only ones achieving a 100% of accuracy (see also Fig. 3) on the images in the test set (see Table 5). It is worth noting that the addition pooling generates lower performance in terms of accuracy (see Table 5 and Fig. 2). The result is in line with the considerations reported in [20].

Table 5: Comparison of the different models during the test phase. We measured the memory (column *Memory*) and CPU (column *CPU*) consumption. Moreover, we computed the accuracy of the models on the images belonging to both validation set (column *Val Acc*) and test set (column *Test Acc*). The former data have been achieved by averaging 10 replications of the training process (data in squared brackets indicate the standard deviation), while the latter refer to the best model found during training. Bold values identify the best results.

| Model | Memory (%) | CPU (%) | Val Acc (%) | Test Acc (%) |
|---|---|---|---|---|
| ConvNet1_2 | 2.063 [0.033] | 49.979 [2.17] | 94.936 [14.712] | 98.5 |
| ConvNet1_2+pool | 1.06 [0.0] | 49.982 [8.397] | 85.073 [22.805] | 95.25 |
| ConvNet2_3 | 2.128 [0.006] | 49.972 [2.346] | **100.0 [0.0]** | 97.25 |
| ConvNet2_3+pool | **0.92 [0.0]** | **49.912 [10.647]** | **100.0 [0.0]** | 94.75 |
| ConvNet3_4 | 2.77 [0.0] | 49.986 [1.543] | **100.0 [0.0]** | **100.0** |
| ConvNet3_4+pool | **0.92 [0.0]** | 49.919 [10.585] | **100.0 [0.0]** | **100.0** |

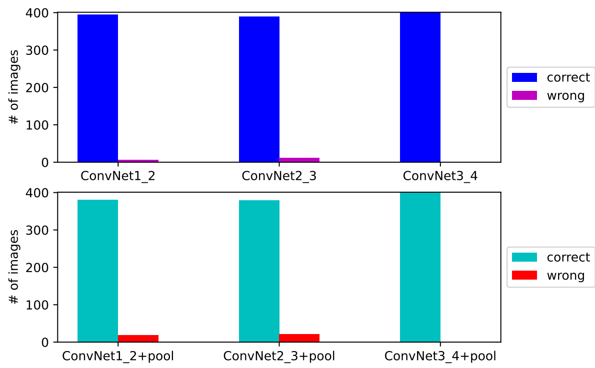


Figure 2: Number of images in the test set correctly and wrongly classified by the models without pooling (top) and with pooling (bottom). The ConvNet3_4 and ConvNet3_4+pool models obtain an accuracy of 100% (see also Fig. 3), while the other models do not manage to correctly categorize all the images.

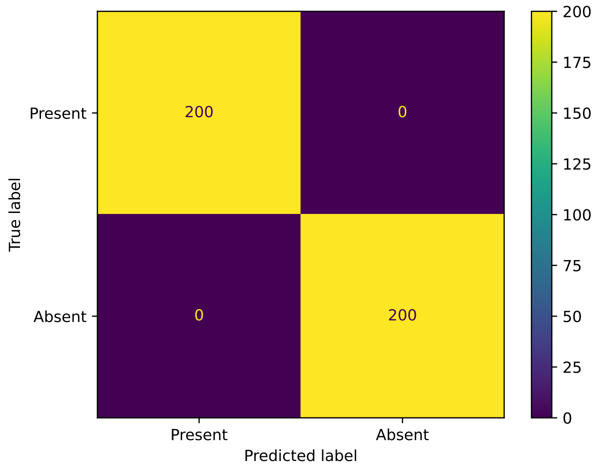


Figure 3: Confusion matrix indicating the capability of the ConvNet3_4 and ConvNet3_4+pool models of categorizing the 400 images in the test set. Data in the diagonal mean correct classifications, while data outside the diagonal indicate wrong classifications. As it can be seen, both models succeed in correctly classifying all the images.

Finally, we tested the models in a real acquisition scenario. Specifically, to simulate the industrial monitoring application to which the system will be applied, we utilized a prototype setup consisting of a Raspberry Pi 4 board with a quad-core processor and 4 GB of RAM, connected to a power supply with an integrated power meter. The Raspberry Pi was connected to a USB camera featuring an 5 MP sensor. In the application, the camera is activated, an image is captured, which is subsequently processed (primarily resizing operations), and finally fed into the model under examination. This acquisition and classification process is repeated every 2 seconds. We measured the power consumption (in watts) and the time required to classify an image (in seconds) for each model (see Table 6, columns *Power* and *Classification time*, respectively). The classification time corresponds to the interval needed by the classifier to make the prediction, excluding the time required to capture the image. We computed the correlation between the model size (see Table 6, column *Size*), the energy consumed during the process, and the time required for classification. We found a strong positive correlation between model size and consumption of energy (Spearman test, $rho = 0.985611$, significant at $p < 0.01$), and a perfect correlation between model size and classification time (Spearman test, $rho = 1.0$, significant at $p < 0.01$). This implies that the heavy models require a remarkable higher amount of resources, which results in a lower sustainability.

Concerning the classification time, Fig. 4 shows the differences between the analyzed models. With the exception of the ConvNet2_3+pool and ConvNet3_4+pool models, which require similar times (Mann-Whitney U test, $p > 0.05$), there are significant differences between the other networks (Mann-Whitney U test, $p < 0.01$ for all compared pairs). Overall, our outcomes reveal how the addition of pooling is undoubtedly beneficial for sustainability. However, the accuracy of such lightweight models is inferior. Therefore, a trade-off between sustainability and accuracy is desirable in order to cope with both the industrial and environmental requirements. The proposed ConvNet3_4+pool model represents a good compromise in this context, since it enhances sustainability with no impact on the final performance.

Table 6: Comparison of the different models during the test in the real scenario. Best values are highlighted in bold (the lower, the better).

| Model | Size (MB) | Power (W) | Classification time (s) |
|---|---|---|---|
| ConvNet1_2 | 633.6 | 3.89 | 1.7 |
| ConvNet1_2+pool | 69.7 | 3.1 | 0.27 |
| ConvNet2_3 | 627.3 | 3.74 | 1.33 |
| ConvNet2_3+pool | 7.4 | **2.85** | 0.18 |
| ConvNet3_4 | 931.5 | 4.0 | 2.86 |
| ConvNet3_4+pool | **1.1** | **2.85** | **0.15** |

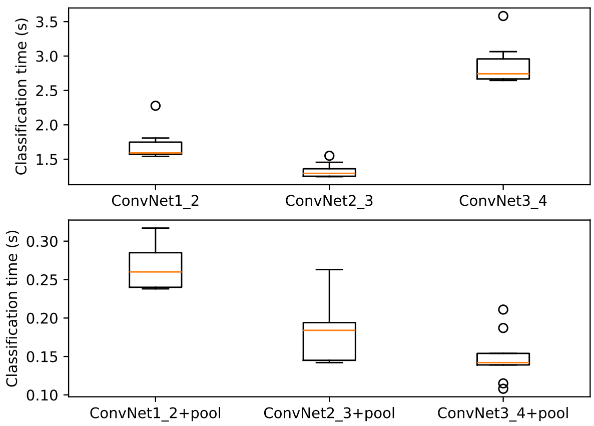


Figure 4: Classification time of the different models without (top) and with (bottom) pooling layers. Boxes represent the inter-quartile range of the data. The red line inside each box marks the median value. The whiskers extend to the most extreme data points falling within 1.5 times the inter-quartile range from the box. Data obtained by replicating the acquisition 9 times.

## 4 CONCLUSIONS

The undeniable impact of human activity on environment raises pressure on the need of changing the way of designing and implementing industrial process in order to enhance sustainability and cost-efficiency, while minimizing the waste of materials and the effect of pollution. In this respect, Industry 4.0 has emerged as a new alternative to classic industrial paradigms, which have demonstrated their inability to cope with environmental concerns, with a significant amount of wasted resources and not negligible disposal costs. For instance, process monitoring is often performed by a human operator, whose activity might be influenced by physical (e.g., fatigue or illness) and/or mental (e.g., stress) and/or external (e.g., lighting) factors. This might generate errors in the control of production lots. Using advanced tools, like AI, may help mitigate such issues and improve sustainability.

In this work, we propose a novel framework to automate the monitoring of transparent tubes inside the production process of a company dealing with plastic consumables. The aim is to improve efficiency and sustainability, while reducing disposal costs. In particular, we addressed an extended version of the problem firstly introduced in [19], with a focus on sustainability. Specifically, we compared the recently introduced ConvNet3_4 model [19,20] with its lightweight version (named ConvNet3_4+pool) implemented with the addition of pooling layers in the convolutional part of the network. We measured the resources required by both models during training and test. The results we obtained indicate that, despite either models achieve a 100% accuracy in both phases, the ConvNet3_4+pool model is remarkably superior in terms of sustainability: it enables to save time, memory, and to some extent CPU. These properties turn out to be paramount when applied to real industrial scenarios. Furthermore, the comparison with other simpler deep networks reveals how the ConvNet3_4+pool model represents the best trade-off between sustainability and effectiveness in the considered context.

In the future, we plan to further analyze the environmental impact of such models. For instance, we are planning to estimate the energy consumption of the presented models using the *energyusage* library [28], which allows to calculate also the carbon emissions. Moreover, as we pointed out in the analyses reported in Section 3, in the considered scenario the addition of pooling provides a remarkable benefit in terms of sustainability at the cost of a slightly lower accuracy. Future research should investigate ways to improve the performance of such models by limiting the amount of utilized resources in order to preserve sustainability. Lastly, we plan to extend the applicability of the proposed models in other domains, like the preliminary study we made in the context of lateral-flow tests [29], with a view at making those processes more sustainable.

## REFERENCES

[1] Goudie, A. S. 2018. “Human impact on the natural environment.” John Wiley & Sons.

[2] Wyns, A. 2023. “COP27 establishes loss and damage fund to respond to human cost of climate change.” The Lancet Planetary Health, 7(1), e21-e22.

[3] Anwar, A., Ahmad, N., and Madni, G. R. 2020. “Industrialization, freight transport and environmental quality: evidence from belt and road initiative economies.” Environmental Science and Pollution Research, 27(7), 7053-7070.

[4] Opoku, E. E. O., and Boachie, M. K. 2020. "The environmental impact of industrialization and foreign direct investment." Energy Policy, 137, 111178.

[5] Yang, B., and Usman, M. 2021. "Do industrialization, economic growth and globalization processes influence the ecological footprint and healthcare expenditures? Fresh insights based on the STIRPAT model for countries with the highest healthcare expenditures." Sustainable Production and Consumption, 28, 893-910.

[6] Brundtland, G. H., and Khalid, M. 1987. "Our common future." Oxford University Press, Oxford, GB.

[7] Chen, X., Despeisse, M., and Johansson, B. 2020. "Environmental sustainability of digitalization in manufacturing: A review." Sustainability, 12(24), 10298.

[8] Daoutidis, P., Zachar, M., and Jogwar, S. S. 2016. "Sustainability and process control: A survey and perspective." Journal of Process Control, 44, pp. 184-206.

[9] Mani, M., Lyons, K. W., Rachuri, S., Subrahmanian, E., Ameta, G., Sriram, R. D., and Feng, S. C. 2008. "Introducing sustainability early into manufacturing process planning." In: Proceedings of the International Manufacturing Science and Engineering Conference (Vol. 48517, pp. 319-326).

[10] Kagermann, H., Lukas, W.-D., and Wahlster, W. 2011. "Industrie 4.0: Mit dem Internet der Dinge auf dem Weg zur 4. industriellen Revolution." VDI nachrichten, 13(1), pp. 2-3.

[11] Enyoghasi, C., and Badurdeen, F. 2021. "Industry 4.0 for sustainable manufacturing: Opportunities at the product, process, and system levels." Resources, Conservation and Recycling, 166, 105362.

[12] Jamwal, A., Agrawal, R., and Sharma, M. 2022. "Deep learning for manufacturing sustainability: Models, applications in Industry 4.0 and implications." International Journal of Information Management Data Insights, 2(2), 100107.

[13] Kamble, S. S., Gunasekaran, A., and Gawankar, S. A. 2018. "Sustainable Industry 4.0 framework: A systematic literature review identifying the current trends and future perspectives." Process Safety and Environmental Protection, 117, pp. 408-425.

[14] Oláh, J., Aburumman, N., Popp, J., Khan, M. A., Haddad, H., and Kitukutha, N. 2020. "Impact of Industry 4.0 on environmental sustainability." Sustainability, 12(11), 4674.

[15] Jan, Z., Ahamed, F., Mayer, W., Patel, N., Grossmann, G., Stumptner, M., and Kuusk, A. 2023. "Artificial intelligence for industry 4.0: Systematic review of applications, challenges, and opportunities." Expert Systems with Applications, 216, 119456.

[16] Park, S. H., Lee, K. H., Park, J. S., and Shin, Y. S. 2022. "Deep learning-based defect detection for sustainable smart manufacturing." Sustainability, 14(5), 2697.

[17] Bochkovskiy, A., Wang, C. Y., and Liao, H. Y. M. 2020. "YOLOv4: Optimal speed and accuracy of object detection." arXiv preprint arXiv:2004.10934.

[18] Walk, J., Kühl, N., Saidani, M., and Schatte, J. 2023. “Artificial intelligence for sustainability: Facilitating sustainable smart product-service systems with computer vision.” Journal of Cleaner Production, 402, 136748.

[19] Zribi, M., Pagliuca, P., and Pitolli, F. 2023. “Convolutional neural networks for the automatic control of consumables for analytical laboratories.” In: Book of Abstract of the BUILD-IT2023 Worskhop, pp. 95-97

[20] Zribi, M., Pagliuca, P., and Pitolli, F. 2024. “A Computer Vision-Based Quality Assessment Technique for the automatic control of consumables for analytical laboratories.” Expert Systems with Applications, 256.

[21] Goodfellow, I., Bengio, Y., and Courville, A. 2016. “Deep learning.” MIT press.

[22] Shorten, C., and Khoshgoftaar, T. M. 2019. “A survey on image data augmentation for deep learning.” Journal of Big Data, 6(1), 1-48.

[23] Tanner, M. A., and Wong, W. H. 1987. “The calculation of posterior distributions by data augmentation.” Journal of the American Statistical Association, 82(398), 528-540.

[24] `https://pytorch.org/vision/main/auto_examples/plot_transforms.html`

[25] Arlot, S., and Celisse, A. 2010. “A survey of cross-validation procedures for model selection.” Statistics Surveys, 4, 40-79

[26] Schaffer, C. 1993. “Selecting a classification method by cross-validation.” Machine Learning, 13, pp. 135-143.

[27] Yates, L. A., Aandahl, Z., Richards, S. A., and Brook, B. W. 2023. “Cross validation for model selection: a review with examples from ecology.” Ecological Monographs, 93(1), e1557.

[28] `https://pypi.org/project/energyusage/`

[29] Tufo, G., Zribi, M., Pagliuca, P. and Pitolli, F. 2024. “An Explainable Convolutional Neural Network for the Detection of Drug Abuse.” In: CEUR Proceedings of the 1st Workshop on Explainable Artificial Intelligence for the Medical Domain (EXPLIMED).