



# From Tables to Computer Vision: Transforming HPDC Process Data into Images for CNN-Based Deep Learning

A. Burzyńska 

University of Warmia and Mazury in Olsztyn, Poland

Corresponding author: E-mail address: [alicja.burzynska@uwm.edu.pl](mailto:alicja.burzynska@uwm.edu.pl)

Received 01.01.2025; accepted in revised form 05.02.2025; available online 30.06.2025

## Abstract

This paper proposes a methodology for leveraging convolutional neural networks (CNNs) in conjunction with advanced data preprocessing to facilitate optimal quality control decision-making in high pressure casting (HPDC) processes. The approach assists in predicting key values of the dependent variable associated with defect occurrence, enabling foundries to enhance product quality, reduce waste, and augment overall production process efficiency. The proposed study is founded on two principal pillars: the transformation of process tabular data (generated using the Conditional Tabular Generative Adversarial Network (CTGAN)), involving the mapping of features onto a fixed grid in a heatmap structure, and the configuration of the CNN algorithm to extract complex patterns in the data that are not readily apparent in the original tabular format. The study utilized a substantial dataset with a total of 61,584 images, and the most effective model attained an impressive Root Mean Square Error (RMSE) of 0.81, underscoring the model's remarkable capacity to accurately detect and predict casting quality issues. The model's efficacy was evaluated through its application to both large and small, differently distributed data sets. Utilizing a combination of statistical pre-processing, intelligent generative models, visual data transformations and deep learning, the methodology offers a comprehensive approach to enhancing production efficiency, ensuring superior process control and improving the quality of HPDC products. This development signifies a significant advancement in the field of intelligent systems for manufacturing process optimization, aligning with the principles of Industry 4.0 and Quality 4.0.

**Keywords:** Computer vision in foundry, Quality 4.0., CNN for Tabular Data, Industry 4.0

## 1. Introduction

The concept of Industry 4.0 has catalyzed a paradigm shift in the domain of manufacturing, resulting in the integration of cutting-edge technologies to achieve unprecedented levels of efficiency, quality, and productivity [1, 2]. Quality 4.0, a transformative concept, lies at the heart of this revolution, synergizing digital tools, data analytics, and artificial intelligence to redefine product quality assurance [3]. This evolution has given rise to smart manufacturing and smart factories, which are sophisticated ecosystems where interconnected systems, real-time data streams,

and autonomous decision-making converge to tackle the intricate challenges of modern industrial processes [4]. These advancements are tailored to navigate escalating complexities while upholding stringent quality benchmarks and promoting sustainable practices.

The field of deep learning, a subfield of artificial intelligence, plays a critical role in the Quality 4.0 paradigm [5]. The employment of deep learning algorithms, such as CNNs, has led to significant advancements in the realm of advanced analytics and predictive modeling [6]. Convolutional neural networks have gained prominence for their proficiency in extracting latent patterns from voluminous data sets, which has historically been challenging



to analyze. The adaptation of CNNs from the domain of image and video analysis to that of industrial processes offers substantial potential in enhancing quality control within manufacturing operations, thus contributing to enhanced efficiency and reliability.

The field of computer vision, an integral component of deep learning, plays a pivotal role in process optimization and defect detection [7]. It empowers the analysis of visual data to identify anomalies, effectively monitor operations, and enhance decision-making processes. Although the primary focus of computer vision methodologies has historically been in the domain of visual inspection tasks, recent advancements have sparked a surge in interest in expanding these techniques to the realm of tabular data industrial processes [8]. The process involves the transformation of structured data into image-like representations, enabling the utilization of convolutional neural networks for pattern recognition and predictive analytics [9].

In the domain of HPDC process a pivotal manufacturing process for the production of complex and precise metal components, ensuring the maintenance of product quality assumes paramount importance [10]. The HPDC process encompasses the rapid injection of molten metal into a mold, characterized by elevated pressure levels and velocity. This results in the formation of components that boast exceptional dimensional accuracy and surface quality. High-pressure die casting is a complex process influenced by numerous factors that critically impact the quality of the final cast product [11]. Nonetheless, the process exhibits vulnerabilities that lead to quality concerns such as porosity, surface imperfections, and dimensional inconsistencies, attributable to fluctuations in process parameters. Conventional statistical methodologies often prove inadequate in capturing the intricate interrelationships between these parameters, underscoring the necessity for the adoption of sophisticated data-driven methodologies.

A notable challenge arises from applying computer vision methods to HPDC process data, which involves transforming the typically tabular process data representing sensor readings, operational parameters and material properties into heatmaps. The generation of these heatmaps enables the utilization of CNNs to identify subtle process anomalies and predict product quality metrics with greater precision than conventional methods.

The scarcity of real-world HPDC process data, attributable to concerns regarding proprietary information, limitations in data collection capabilities, and considerable experimental costs, introduces an additional layer of complexity. To address this challenge, synthetic data generation techniques, such as the CTGAN [12], offer a promising solution. By generating representative synthetic datasets, these methods enable robust model training and evaluation while preserving the statistical properties of the original data [13]. The generation of synthetic data facilitates reproducibility and scalability in research, thus overcoming the limitations of small or incomplete datasets.

Researchers across various disciplines like medicine [14, 15], bioinformatics [16, 17, 18] and others [19-27], have investigated similar challenges involving non-image data transformation, processing, feature extraction and pattern recognition, thereby underlining the pertinence of this subject. In certain cases, tabular data can be reorganized in a 2D space to explicitly highlight relationships between features, such as categories or similarities among them [28, 29, 30]. To date, four primary methods have been

proposed for transforming non-image tabular data into image representations suitable for predictive modelling with CNNs. REFINED (REpresentation of Features as Images with NEighborhood Dependencies), which uses Bayesian multidimensional scaling to globally minimize distortion while projecting features into 2D [29]. The projected features are then mapped to image pixels, and a hill-climbing algorithm is applied to further refine the arrangement of feature positions in the image [29]. DeepInsight, which uses t-SNE to project feature vectors into a 2D space by minimizing the Kullback-Leibler divergence between the feature distributions in the original high-dimensional space and the 2D projection [28]. Within this projection, a minimal rectangular region encompassing all feature points is identified, forming the image representation [31]. OmicsMapNet, which is specifically designed for gene expression data [30]. This method transforms cancer patient data into 2D images for tumor grade prediction by using functional gene annotations from the Kyoto Encyclopedia of Genes and Genomes (KEGG) [32]. Using the TreeMap approach, OmicsMapNet ensures that genes with similar molecular functions are positioned close together in the image. Generator for Tabular Data (IGTD), which assign features to pixels by minimizing the difference between two rankings: the ranking of pairwise distances between features in the original data and the ranking of pairwise distances between the assigned pixels in the generated image [33]. The pixel distances are calculated based on their coordinates within the image. Compared to the four existing methods the proposed approach offers several distinct advantages described in chapter 5.

In this study, a novel approach to the prediction of quality in HPDC process is presented. This approach involves the new transformation of tabular process data into image representations and the employment of CNNs for in-depth learning analysis [34]. The study commences with the generation of synthetic HPDC process data using CTGAN, with the quality of this data being validated through the application of statistical metrics and the replication of experiments conducted on real benchmark datasets. Subsequently, the transformation of tabular data into heatmap images is demonstrated, thus enabling their utilization as input for CNN models. Finally, we assess the predictive capabilities of the proposed approach in determining casting quality, highlighting its potential for advancing Quality 4.0 initiatives in smart manufacturing.

## 2. Synthetic Tabular Data Preparation and Evaluation

The generation and evaluation of synthetic tabular data signifies a substantial innovation in overcoming the challenges associated with real-world datasets in the foundry industry, particularly in HPDC process research. Industrial data frequently exhibits imperfections, including poor quality, imbalanced value representation, diverse variable distributions, and correlations between process parameters. These issues complicate the development of machine learning (ML) models, which rely on large, well-structured datasets. To address these challenges, the present study utilized the CTGAN a cutting-edge tool designed for synthesizing tabular data has been shown to be especially effective

in generating datasets that faithfully capture the intricate statistical characteristics of industrial HPDC processes.

The present study utilized the CTGAN model to generate synthetic data, drawing upon information from the benchmark HPDC process dataset (described in [35, 36, 37]). The model, implemented in Python code, was initiated with default parameters, which included a generator and discriminator learning rate of 0.0002 and dimensions set at 128 [38]. The primary objective of the model was to learn statistical distributions and relationships in tabular data, with the aim of generating synthetic samples that exhibited comparable characteristics. The model was fitted to the data using the 'fit(df, epochs=300)' method, which involved training for 300 epochs, enabling the generator and discriminator to optimize the loss function collaboratively [38]. The generator then generated synthetic samples that were evaluated by the discriminator as true or false, with both components learning through a competitive process. Following the training phase, the model was configured to generate synthetic data using the sample(n=len(df)) method. In this instance, the number of rows generated corresponded to the benchmark dataset [35, 36, 37]. The created dataset consisted of 58 variables and more than 10,000 samples of analyzed castings. The dependent variable, 'leakage in the high-pressure circuit' served as a direct indicator of porosity linked with casting quality. Independent variables encompassed process parameters potentially affecting leakage, forming the foundation for further analysis.

The fidelity of the synthetic data was then validated by conducting a series of comparative analyses against the aforementioned benchmark dataset, utilizing a range of statistical metrics including the mean, standard deviation, minimum and maximum values, and quartiles (25%, median, 75%) [39]. The relative differences (RD) [40] between the descriptive statistics of the synthetic data (SV) and the benchmark values (BV) were evaluated by computing them using equation 1. Visualization techniques, including histograms and run charts, provided insights into the alignment of data. The Synthetic Data Quality Score (SDQS) [41] has been developed as a metric for the evaluation of the alignment of synthetic datasets with expected statistical properties. The calculation of this score, as outlined in equation 2, involves the averaging of the relative differences across all metrics evaluated, with the results expressed as a percentage. In this formula, k denotes the number of statistical measures, and  $RD_{S_i}$  corresponds to the average relative difference for the i-th statistic. By aggregating these differences, the SDQS offers an efficient and precise evaluation of the statistical similarity between the synthetic data and the benchmark values.

$$RD = \frac{|SV - BV|}{|BV|} \times 100 \quad (1)$$

$$SDQS = \frac{1}{k} \sum RD_{S_i} \quad (2)$$

The computed SDQS demonstrated a substantial alignment with a score of 4.43%, thereby validating the quality of the synthetic data. The findings underscore the fact that synthetic data meticulously adheres to the anticipated standards, thereby affirming its viability for analytical pursuits and pragmatic applications and corroborating its aptitude for sophisticated modelling.

The study was built on the preceding foundation described in [35, 36, 37] by means of a comprehensive statistical framework integrating analysis of variance direct and reversed (ANOVA) and the Kruskal-Wallis test (K-W) direct and reversed (RK-W). This was employed so as to identify critical process variables affecting casting defects on the basis of F, H and p-value statistics. The data was divided into five distinct subsets according to defect severity. The first research set included the complete synthetic dataset of 10 094 observations generated by the CTGAN model without filtering. The second set with 90 observations focused on cases with higher leakage levels, containing records where the dependent variable was  $\geq 8$ . The third set with 10 094 observations represented lower leakage levels, including records where the dependent variable was  $< 8$ . The fourth set with 180 observations combined all data from the second set with 90 records from the upper range of the third set, while the fifth set with 160 observations integrated the second set with 90 records from the third set, selected at equal record intervals. In order to assess variable significance, three selection criteria were used K-W, RK-W and ANOVA. The relevance of the input features was refined progressively by each of these.

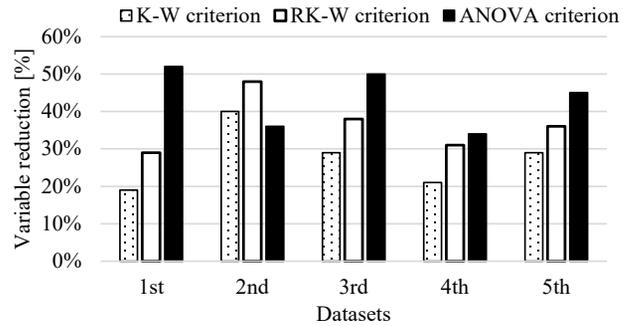


Fig. 1. Results of data dimensionality reduction

Fifteen datasets were constructed by combining variable selection strategies with tailored data subsets. This iterative approach reduced redundancy while also enhancing the interpretability of key influences on casting quality. The consequence of this was a 52% reduction in the number of input variables in the data dimensionality reduction step (presented in Figure 1.). This result is less favorable when compared to that obtained on the benchmark data [35, 36, 37]. The upshot of this is a more complex data analysis and modelling process.

### 3. Tabular-to-Image Transformation

The conversion of tabular data as the most common type of data in industry [42, 43] and foundry industry into image representations constitutes a pioneering linkage between structured datasets and computer vision techniques, thereby enabling the application of CNNs to traditionally non-visual data domains. This conversion involves the conversion of tabular process parameters into grid-based visual structures, thereby facilitating the detection of intricate patterns and dependencies that might otherwise go unnoticed by conventional statistical methods [44]. The transformation of tabular data into image representations constituted a pivotal innovation in the field of data representation,

offering a framework for advanced pattern recognition and process analysis through the utilization of visual representation [34]. This transformation was implemented programmatically to ensure consistency and reproducibility, whilst leveraging Python's robust computing libraries. The flowchart (presented in Figure 4.) and algorithm steps are presented as follows [19]:

Data preparation:

- Step 1: Load input data
  - Load Excel files from the specified directory.
  - Extract data into a pandas data frame.
- Step 2: Prepare output directory
  - Check if the output directory exists.
  - Create the directory if it does not exist.
- Step 3: Separate independent and dependent variables
  - All columns (independent variables) except the first one normalize to the range  $<0, 1>$  and treat as features.
  - The first (dependent variable) column treat as the label.
- Step 4: Determine grid dimensions
  - If the number of independent variables is in the range  $36 < n < 47$ , set grid dimensions to  $6 \times 6$ .
  - If the number of variables is in the range  $< 37, 49 >$ , set grid dimensions to  $7 \times 7$ .
- Step 5: Iterate over normalized rows
  - Reshape the row into the determined grid dimensions.
  - Use zero padding if necessary to fill the grid.
  - Generate a heatmap visualization.
  - Save the heatmap as a .png file in the output directory with name related to label.

The process commenced with the normalization of independent variables, utilizing the 'MinMax scaling' method from the 'sklearn.preprocessing' library, which rescaled values to a range of  $<0-1>$ . This ensured the preservation of the relative relationships among variables while standardizing their magnitudes for consistent visualization. The dependent variable, representing the quality metric, was preserved as a label, ensuring compatibility with downstream supervised learning tasks.

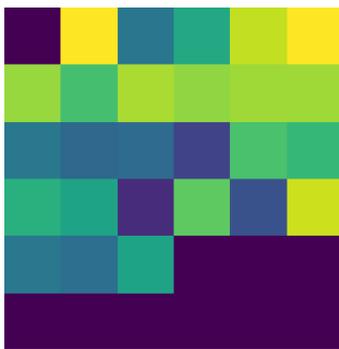


Fig. 2. Results of conversion of one tabular observation from 1<sup>st</sup> research set according to K-W criterion to  $6 \times 6$  image

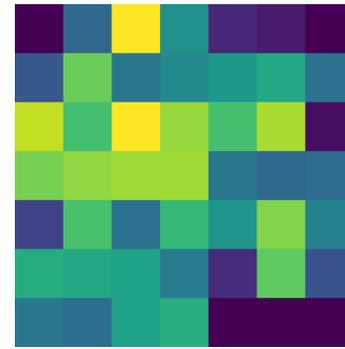


Fig. 3. Results of conversion of one tabular observation from 1<sup>st</sup> research set according to ANOVA criterion to  $7 \times 7$  image

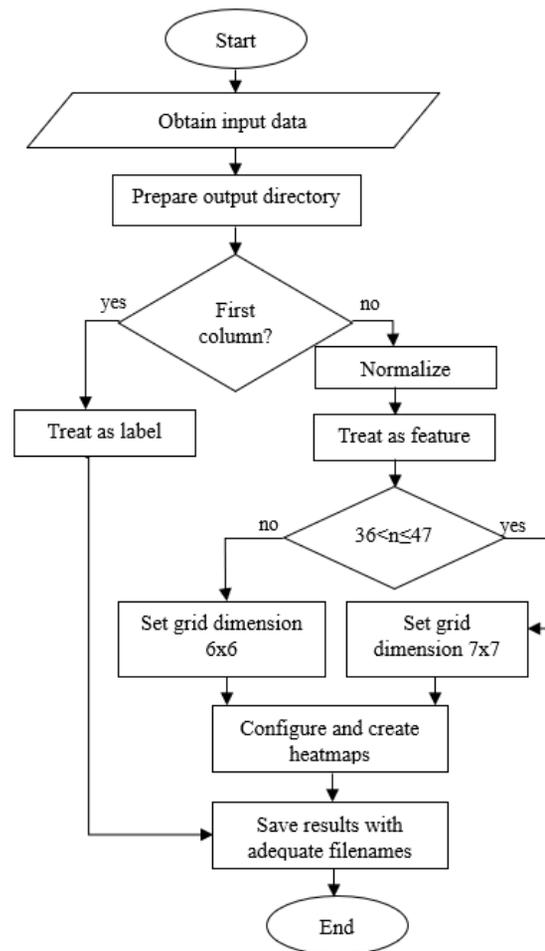


Fig.4. Flowchart of the tabular data transformation into heatmap visualization process

The transformation of the tabulated data into a two-dimensional grid structure was performed for each row of data corresponding to a single observation [28]. The grid dimensions were set to  $6 \times 6$  when the number of variables ( $n$ ) did not exceed 36 (presented in Figure 2.), or  $7 \times 7$  [44] when the number of dependent variables did not exceed 49 (presented in Figure 3.). In instances

where the number of variables was inadequate to fill the entire grid, zeros were introduced to ensure the grid remained constant in size.

The creation of heatmaps from these grids was achieved using Python's matplotlib library. For each observation, a grid of normalized data was rendered into a heatmap where the intensity of each grid cell represented the corresponding variable's value. The "viridis" colormap was selected for its perceptually uniform gradient and high contrast, enabling clear visual distinction between variable intensities. Visual clutter was minimized by omitting all non-essential elements, such as axes, labels, and gridlines, resulting in clean, computationally efficient image outputs.

In the created heatmap, cells with values close to 0 were depicted in a deep shade of blue, whereas those with values closer to 1 appeared in yellow [34]. This color gradient effectively represented the normalized value range. The blue tones highlighted lower values, including zero, whereas yellow emphasized the higher values, thus providing a comprehensive representation of the data variations across the heatmap. The intermediate values were displayed in green and teal shades, ensuring a clear and easily comprehensible visualization of the data variation. This systematic mapping of variables to specific grid cells preserved contextual relationships and ensured consistency across observations.

The workflow encompassed automated file management, wherein the heatmaps were saved as high-resolution 61 584 PNG files in a predefined directory. The filenames encoded key metadata, including dataset identifiers, observation indices, and dependent variable labels, ensuring traceability and facilitating downstream analysis. The systematic iteration through observations ensured that every row of tabular data was transformed into its corresponding heatmap, thereby creating a comprehensive visual representation of the dataset.

## 4. CNN-Based Deep Learning Quality Prediction Model

The development of a CNN-based model for predicting quality metric, so value of the dependent variable in HPDC processes constituted a pivotal component of the present study. The model with adequate parameters (shown in Table 1.), was applied to each of the 15 datasets (shown in Table 2.), implementing two distinct data handling methodologies: a 100% split for training and prediction on the entire dataset, and an 80% split for training and 20% testing. CNN architecture, customized for these two scenarios, exhibited both robustness and adaptability in the analysis of heatmap representations of tabular data. The flowchart (presented in Figure 5.) and algorithm steps of the proposed methodology are presented as follows [19]:

Data preparation:

- Step 1: Load Input Images and Labels
  - Load image files from the specified directory
  - Extract labels from filenames.
  - Resize images to either 6x6 or 7x7 pixels with 3 channels (RGB).
- Step 2: Normalize image data
  - Scale the pixel values to the range <0, 1>.
- Step 3: Convert images and labels into NumPy arrays

- Convert image data into a 4D array: (number\_of\_images, height, width, channels).
  - Convert the labels to a 1D array: (number\_of\_images,).
- Dataset splitting:
- Step 4: Split dataset into training and testing sets
    - Use an 80/20 split for selected datasets.
    - For other cases, use a 100/0 split (no separate test set).
- Architecture definition:
- Step 5: Define the CNN architecture
    - Add the convolutional layer.
    - Add a max pooling layer.
    - Add a flatten layer.
    - Add a dense layer.
    - Add an output dense layer.
- Model compilation:
- Step 6: Compile the model
    - Use the Adam optimizer.
    - Use MSE as the loss function.
    - Use RMSE as the performance metric.
- Model training:
- Step 7: Train the model
    - Set the batch size (32 for big datasets, 8 for smaller).
    - Train for up to 400 epochs using early stopping (patience = 15).
- Model evaluation:
- Step 8: Evaluate the model
    - Predict values of the dependent variable
    - Calculate evaluation metrics: MSE, RMSE, and MAE.
- Results and visualization:
- Step 9: Save results and generate plots
    - Save training history, evaluation metrics, and predictions to Excel files.
    - Generate plots: training loss curve, scatter plot for true vs. predicted values, residual plot for errors.
- Output storage:
- Step 10: Save outputs to the results directory.

Table 1.

Model parameters description

Parameter	Description
Input shape	(6, 6, 3) or (7, 7, 3)
Normalization	Pixel values scaled to <0, 1> by dividing by 255
Optimizer	Adam
Train/Test split	100/0 or 80/20
Loss Function	MSE
Metrics	MAE
Batch size	8 or 32
Epochs	400
Early Stopping	Loss with patience 15

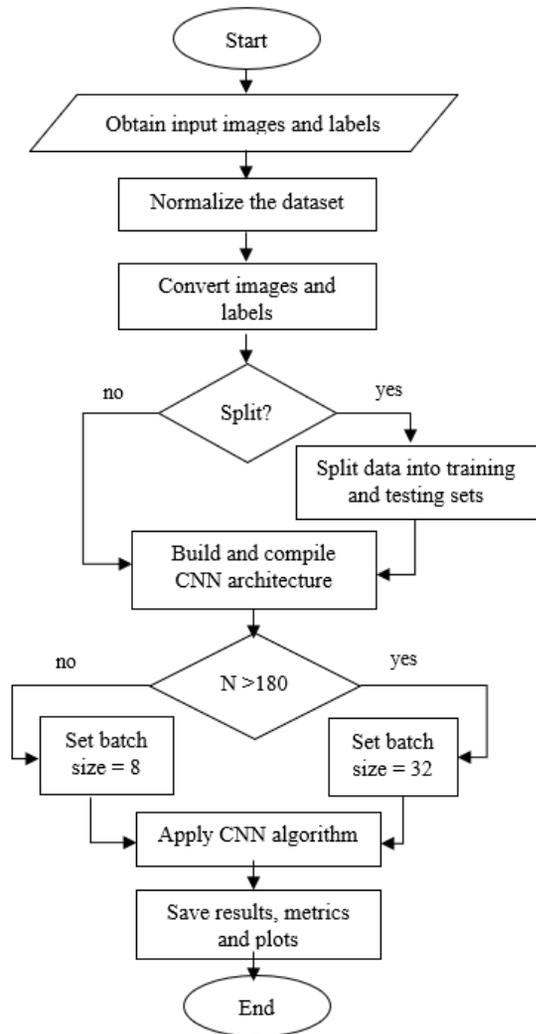


Fig. 5. Flowchart of CNN-based methodology for prediction values of dependent variable

Table 2. Datasets features description

Model nb	Dataset nb / Criterion	Total size	Image dimension	Train/test set size [%]	Label range
1	1/K-W	10094	6x6	100/0	0,01 -
2				80/20	222,27
3	1/RK-W	10094	7x7	100/0	0,01 -
4				80/20	222,27
5	1/ANOVA	10094	7x7	100/0	0,01 -
6				80/20	222,27
7	2/K-W	90	6x6	100/0	8,00 -
8				80/20	222,27
9	2/RK-W	90	6x6	100/0	8,00 -
10				80/20	222,27
11	2/ANOVA	90	6x6	100/0	8,00 -
12				80/20	222,27

13	3/K-W	10004	6x6	100/0	0,01 -
14				80/20	7,99
15	3/RK-W	10004	6x6	100/0	0,01 -
16				80/20	7,99
17	3/ANOVA	10004	7x7	100/0	0,01 -
18				80/20	7,99
19	4/K-W	180	7x7	100/0	7,33 -
20				80/20	222,27
21	4/RK-W	180	7x7	100/0	7,33 -
22				80/20	222,27
23	4/ANOVA	180	7x7	100/0	7,33 -
24				80/20	222,27
25	5/K-W	160	6x6	100/0	0,01 -
26				80/20	222,27
27	5/RK-W	160	6x6	100/0	0,01 -
28				80/20	222,27
29	5/ANOVA	160	7x7	100/0	0,01 -
30				80/20	222,27

The methodological steps mentioned above indicate that the CNN was designed to process input images with dimensions of 7x7 and 6x6 pixels, corresponding to the heatmaps derived from the normalized tabular data. The network's architecture (presented in Figure 6.), consisted of convolutional layers, a flattening layer and dense layers. The first convolutional layer featured 32 filters with a kernel size of 3x3, designed to extract spatial features from the input images [6]. This was followed by a rectified linear unit (ReLU) activation function to introduce non-linearity. A MaxPooling layer, with a pool size of 2x2, reduced the spatial dimensions, thus retaining essential features while minimizing computational complexity [45, 46]. Subsequent to the convolutional and pooling layers, the feature maps were flattened into a one-dimensional vector, thereby preparing the data for the fully connected layers. The network included one dense layer with 64 neurons and a ReLU activation function to process the extracted features. The output layer contained a single neuron with a linear activation function, suitable for the regression task of predicting continuous quality metrics [47].

The model was trained using the Adam optimizer, which offered adaptive learning rates to enhance convergence [48]. The loss function was Mean Squared Error (MSE), which is suitable for regression tasks, and the Root Mean Square Error (RMSE) was calculated to assess the overall deviation of predictions from true values. This provided a robust measure of model performance. Additionally, the training process was monitored using the Mean Absolute Error (MAE) metric, which offers a straightforward indication of prediction accuracy [49].

In scenarios where the entirety of the dataset was utilized for training and prediction. Initially, all images and their corresponding labels were loaded. Subsequently, the image data was normalized to scale pixel values within the range  $\langle 0, 1 \rangle$ . The model was then trained on the complete dataset without the reservation of a test set. The maximum number of epochs allowed was 400. Early stopping, based on the loss function, was implemented to prevent overfitting, and a patience parameter of 15 epochs was employed to ensure the model did not overtrain.

The generation of detailed metrics, including MSE, RMSE, and MAE, was achieved and saved to structured directories. Visual insights into the model's performance were provided by scatter

plots comparing true vs. predicted values and residual plots. For datasets where 80% of the data was allocated for training, and 20% was reserved for testing. In the context of smaller datasets with number (N) variables smaller than 180, it was necessary to reduce the batch size from 32 to 8 in order to accommodate the limited data availability and to ensure the efficient training of the model

[50]. The number of epochs was set based on the loss curve, thus ensuring that the model was trained adequately without overfitting.

The model's performance across the research datasets demonstrated significant variability, influenced by the dataset type. For the datasets with 100% values in the training set, the RMSE values remained consistently low across datasets, indicating high predictive accuracy when the model was trained and evaluated on

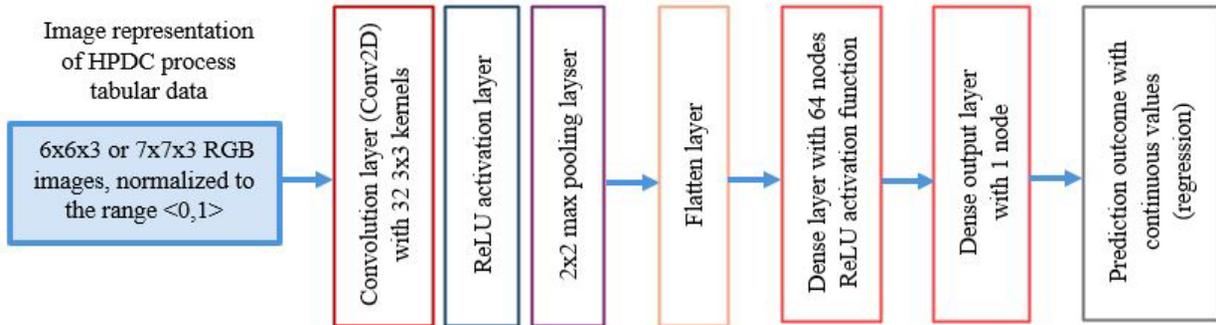


Fig.6. Architecture of the CNN used for predicting 'leakage in the high-pressure circuit' values, based on image representations [37]

the same data (shown in Table 3.). For example, model 1, which used a dataset of 10,094 samples, achieved an average MSE of 2.00, average MAE of 0.96 and average RMSE of 1.41, demonstrating its ability to learn effectively from big datasets. While this approach minimized training error, it risked overfitting due to the lack of a test set to evaluate. Models that were trained with an 80/20 split generally showed more reliable generalization metrics, as the inclusion of a test set provided an unbiased assessment of the performance of the model. For example, Model 2, reflecting a balance between training and testing performance, achieved an average MSE of 8.50 and an average RMSE of 2.91. Similarly, Model 14 achieved low error values (average MSE = 1.24, average RMSE = 1.11) on data sets with smaller label ranges, confirming its ability to generalize effectively.

18	262,51	1,27	0,74	1,12
19	26,37	0,76	0,71	0,87
20	21,17	29,42	2,49	5,35
21	26,53	1,14	0,81	1,04
22	22,91	52,30	2,85	7,00
23	27,16	0,79	0,64	0,81
24	21,18	70,11	2,89	8,34
25	22,45	1,98	1,09	1,37
26	20,68	427,50	5,42	20,67
27	24,83	3,14	1,31	1,69
28	18,79	322,90	5,02	17,90
29	25,03	2,34	1,2	1,50
30	20,31	353,39	5,25	18,78

Table 3. Performance metrics of CNN experimental results

Model nb	Average Processing Time [s]	Average MSE	Average MAE	Average RMSE
1	173,49	2,00	0,96	1,41
2	150,60	8,50	0,94	2,91
3	79,55	2,22	0,94	2,99
4	59,21	10,32	0,97	3,21
5	222,08	1,76	0,87	1,32
6	126,84	8,99	0,96	2,99
7	15,44	2,97	1,36	1,72
8	15,93	310,72	7,36	17,62
9	15,2	23,26	3,68	4,80
10	12,53	394,88	7,27	19,78
11	16,05	9,78	2,30	2,91
12	12,71	474,01	8,12	21,71
13	335,94	0,92	0,68	0,96
14	274,51	1,24	0,74	1,11
15	329,18	1,01	0,69	1,00
16	261,25	1,31	0,76	1,14
17	303,61	1,03	0,72	1,01

Models that were trained on big data sets showed a strong performance in both splits. It also should be noted that the results show significant differences between the splits for smaller datasets. For instance, the performance of model 7 was significantly better than that of model 8. It appears that the reduced quantity of training data in the 80/20 split has had a detrimental effect on performance. This observation highlights the challenge of generalization when a subset of the data is reserved for testing. A comparison of feature selection methods (K-W, RK-W, and ANOVA) was also conducted, revealing that datasets created using the K-W criterion exhibited the greatest robustness across a range of dataset sizes and splits. The ANOVA and RK-W criterion demonstrated excellence for big datasets but exhibited greater variability with small datasets, likely due to its reliance on sufficient sample sizes for feature selection. It was also established that the duration of the processing increases in proportion to the magnitude of the dataset, with the exception of instances where training was terminated prematurely as a consequence of the implementation of an early stopping mechanism. This observation underscores the efficacy of early stopping in averting superfluous computational overhead by bringing training to a halt once the model attains convergence, irrespective of the dataset's magnitude.

Scatter plots of true versus predicted values for the datasets with 100% values in the training set showed a strong alignment along the ideal prediction line (presented in Figure 7.), while residual plots revealed minimal systematic errors.

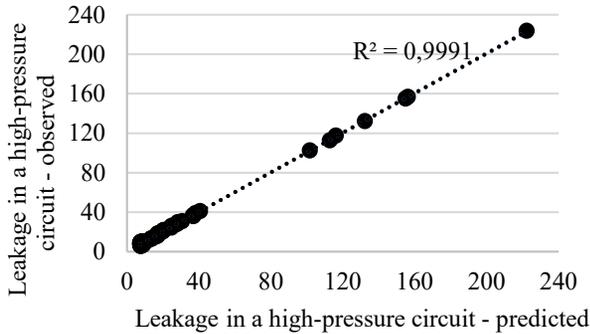


Fig. 7. Results of CNN modelling of the 4th Research Dataset according to K-W criterion with 100% values in the training set

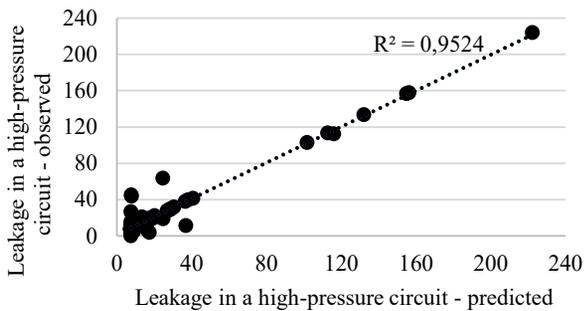


Fig. 8. Results of CNN modelling of the 4th Research Dataset according to K-W criterion with 80% values in the training set

A comparison was also made between the results of image modelling using CNNs and the results of table data modelling by Artificial Neural Networks (ANNs) (shown in Table 4.) based on the methodology described in [35, 36]. It is evident from the results obtained after converting the data into images that lower RMSE values are achieved for the big dataset 1 for K-W and ANOVA criterion and big dataset 3 across all criteria (where a high variability of the value of the dependent variable is observed), additionally for dataset 4 for K-W criterion (whose analysis will allow the most valuable conclusions, on the topic of the influence of specific parameters on the increase of the value of the dependent variable) and 5 (which is the most balanced), indicating high performance in these specific cases. In contrast, ANN achieves low RMSEs for the small dataset 2 across all criteria (in which a high variability in the value of the dependent variable can be observed). The selection of the most suitable model is thus contingent upon the specific application in data driven decision system [51] context and the accessibility of training data.

This CNN-based deep learning framework represents a significant advancement in quality prediction for HPDC process, showcasing its capacity to analyze complex manufacturing datasets and providing actionable insights for process optimization. Future

studies could focus on enhancing generalization through advanced regularization techniques or modelling based on the hybrid datasets.

Table 4. Results of ANN modelling on synthetic datasets

Criterion	Average	Average	Average	Average	Average
	RMSE	RMSE	RMSE	RMSE	RMSE
	Dataset	Dataset	Dataset	Dataset	Dataset
	1	2	3	4	5
K-W	3,12	6,53	1,39	7,27	3,31
RK-W	2,11	0,21	1,35	3,04	4,47
ANOVA	2,12	0,1	1,34	2,47	3,73

## 5. Limitations and Discussion

In this study, an innovative method for analyzing HPDC process data was introduced. This method involves the transformation of non-image data into meaningful heatmap images through a custom algorithm. This transformation enables the use of CNNs for predictive modelling, offering new avenues for data interpretation. However, while this approach leverages the strengths of CNNs, it does not address all challenges associated with HPDC process data. It is therefore vital to acknowledge these limitations in order to provide a contextual framework for our findings and shape future research endeavors.

The proposed methodology in this study employs a substantially more straightforward approach, whereby features are mapped sequentially into a fixed grid, such as 6x6 or 7x7. Presented method can be applied without requiring any prior domain knowledge unlike OmicsMapNet, which relies on domain-specific knowledge. In contradistinction to DeepInsight, REFINED, or IGTD, it does not involve complex dimensionality reduction or optimization processes, which renders it significantly faster and easier to implement. This simplicity ensures image generation is computationally efficient, making the method particularly suitable for real-time or resource-constrained industrial environments, such as HPDC applications. The use of fixed grid dimensions ensures consistency across all generated images, simplifying downstream CNN training and making the method scalable for datasets of varying sizes. However, the straightforward approach also comes with trade-offs. For instance, the technique exhibits an inability to sustain the relationships between features during the mapping process. The allocation of features to the grid is predicated exclusively by their sequence in the dataset, with no consideration given to their correlations or interactions. This absence of optimization results in features with strong relationships that may be positioned at considerable distances from each other in the image, thereby diminishing the CNN's capacity to effectively capture these patterns. The simplified feature mapping process was implemented intentionally, as the tabular data underwent rigorous preprocessing and optimization prior to transformation into image representations. The dataset was refined using ANOVA, K-W, RK-W analysis and Pearson and Spearman correlation analyses to identify and select the most relevant features while minimizing redundancy. This preprocessing step ensured that only the most statistically significant and non-redundant variables were included

in the final dataset, thereby effectively reducing the datasets dimensionality. It is posited that by relying on this rigorous optimization procedure, the method could bypass the more computationally expensive dynamic feature placement strategies employed by methods such as IGTD and REFINED, while still ensuring an adequate representation of the data for CNN modelling. This approach achieves an equilibrium between computational efficiency and data quality, aligning well with the practical requirements of industrial applications.

The CNN method has been shown to have significant potential in regression tasks when applied to HPDC process modelling. While the model demonstrates practical utility, it is important to note that it also faces limitations that influence its generalization and performance. A pivotal element in preventing the model from overfitting the training data is the implementation of early stopping, which has been incorporated. Nevertheless, despite this measure, overfitting can potentially arise in smaller datasets or those exhibiting limited variability. Although early stopping impedes unnecessary overtraining, the absence of supplementary regularization techniques, e.g., dropout or L2 regularization, signifies that the model might still encode patterns specific to the training dataset. The employment of straightforward architecture results in training durations ranging from 15 to 400 seconds, contingent on the size and complexity of the dataset. The proposed model demonstrates notable strengths in terms of computational efficiency. This efficiency renders the model suitable for industrial applications, where time and resource constraints frequently prove to be critical factors. Nevertheless, the simplicity of the architecture may constrain its capacity to model more intricate feature relationships. Another point to consider is the use of MSE as the loss function. While this is appropriate for regression, it should be noted that it introduces sensitivity to outliers, as larger errors disproportionately influence the overall loss. An alternative that could potentially stabilize model performance is to explore the use of Huber loss.

Notwithstanding the aforementioned limitations, the model is suited to its intended application, achieving a balance between simplicity and computational efficiency. By addressing its limitations through dedicated strategies such as feature clustering and alternative loss functions, the model could be further optimized to enhance its robustness and predictive accuracy. Nevertheless, even in its current form, the model may serve as a practical and efficient tool in industrial contexts related to HPDC process quality control.

## 6. Conclusions and Future Directions

This research emphasizes the pivotal function of CNN-based deep learning in transforming quality control within HPDC processes. The ability to predict casting quality with precision is of the utmost importance for the minimization of defects, the reduction of waste, and the enhancement of operational efficiency.

The modelling results confirm the validity of the proposed approach. The lowest RMSE result, 0.44, was achieved on a small dataset – 4, defined according to the ANOVA criterion, using 100% of the data for training. Additionally, this dataset yielded the lowest average RMSE across all trials, equal to 0.81. For large datasets, the lowest RMSE was obtained on Dataset 3, defined according to

the K-W criterion, also using 100% of the data for training. Here, the lowest RMSE was 0.92, with an average RMSE across all trials of 0.96. When the data was split into 80% training and 20% testing, the best RMSE result, 1.09, was also achieved on the large dataset 3, defined by the K-W criterion, yielding the lowest average RMSE for this split at 1.11.

The current drive for solutions that enable rapid analysis to support decision making in industrial environments, while minimizing storage requirements, is a pressing priority. Organizations using AI methods are increasingly required to allocate additional storage in cloud computing environments, with predictions that storage requirements will double within the next three years. While long-term data storage has historically improved the predictive quality of AI-based systems, there is a growing focus on developing algorithms capable of real-time analysis and forecasting without significantly impacting corporate computing resources, particularly in industries such as foundries. The solution proposed in this study is very significant in terms of the above arguments and could contribute to the advancement of such systems and methodologies. Additionally, is limited by the lack of a strategy for spatially clustering related features, which limits the model's ability to identify the critical variables that drive the regression result so predicted values of dependent variable related directly with casting defect occurrence. Therefore, the next step in research development should be to identify the variables that significantly influence the formation of a defect in a casting using DL methods. The following stages of this research should entail the application of alternative methods like REFINED to raw foundry data without prior preprocessing, whilst simultaneously exploring more advanced CNN architectures. In addition, hybrid approaches integrating convolutional and recurrent networks could be investigated to accommodate both the spatial and temporal dynamics inherent in process data. As another potential extension of the current research, it is proposed that the output be redefined as a categorized system of labels, such as "repairable castings," "scrap castings," and "defect-free castings." This would align the results with those of accuracy-based studies. Under this classification scheme, the performance originally assessed via the RMSE can be directly compared to accuracy metrics from similar investigations. This would provide a more comprehensive insight into the performance of the model. These approaches, alongside the one presented in this study, are undoubtedly anticipated and desired by manufacturers aiming to enhance product and castings quality and optimize their production processes.

## References

- [1] Mohan, T.R., Roselyn, J.P., Uthra, R.A. (2022). Digital smart kaizen to improve quality rate through total productive maintenance implemented industry 4.0. In 2022 IEEE 3rd Global Conference for Advancement in Technology (GCAT), 7-9 October 2022 (pp. 1-6). Bangalore, India. DOI: 10.1109/GCAT55367.2022.9971890.
- [2] Hsiao, H., Hung, M., Chen, Ch., Lin, Y. (2022). Cloud computing, internet of things (IoT), edge computing, and big data infrastructure. In Fan-Tien Cheng (Eds.), *Industry 4.1: Intelligent Manufacturing with Zero Defects* (pp.129-167). DOI: 10.1002/9781119739920.ch4.

- [3] Aleksandrova, S.V., Vasiliev, V.A., Alexandrov, M.N. (2019). Integration of quality management and digital technologies. In 2019 International Conference Quality Management, Transport and Information Security, Technologies (IT&QM&IS), 23-27 September 2019 (pp. 20-22). Sochi, Russia. DOI: 10.1109/ITQMIS.2019.8928426.
- [4] Shen, L., Wang, F., Zhou, Z., Xu, R. (2024). Research on the difficulty mining algorithm for the integration of multiple sets of data platforms based on big data analysis in smart factories. In 2024 IEEE 3rd International Conference on Electrical Engineering, Big Data and Algorithms (EEBDA), 27-29 February 2024 (pp. 1444-1448). Changchun, China. DOI: 10.1109/EEBDA60612.2024.10485802.
- [5] Li, D., Chen, X., Tan, P., Jia, J. (2024). How deep learning is implemented in manufacturing quality control. In 2024 3rd International Conference on Artificial Intelligence and Computer Information Technology (AICIT), 20-22 September 2024 (pp. 1-4). Yichang, China. DOI: 10.1109/AICIT62434.2024.10729923.
- [6] Zhao, X., Wang, L., Zhang, Y., Han, X., Deveci, M. & Parmar, M. (2024). A review of convolutional neural networks in computer vision. *Artificial Intelligence Review*. 57(4), 99. DOI: 10.1007/s10462-024-10721-6.
- [7] Islam, R., Zamil, Z.H., Rayed, E. & Kabir, M. (2024). Deep learning and computer vision techniques for enhanced quality control in manufacturing processes. *IEEE Access*. 12, 99, 121449-121479. DOI: 10.1109/ACCESS.2024.3453664.
- [8] Ameri, R., Hsu, Ch., Band, S. (2024). A systematic review of deep learning approaches for surface defect detection in industrial applications. *Engineering Applications of Artificial Intelligence*. 130, 107717, 1-24. DOI: 10.1016/j.engappai.2023.107717.
- [9] O'Shea K., Nash R. (2015). *An introduction to convolutional neural networks*. Retrieved December 15, 2024, from <https://doi.org/10.48550/arXiv.1511.08458>.
- [10] Rahat, A.B. (2024). *High-Pressure die casting (HPDC): precision, efficiency and innovation*. Retrieved December 13, 2024, from <https://www.linkedin.com/pulse/high-pressure-die-casting-hpdc-precision-efficiency-rahat-a-bhatia-vjh2c>.
- [11] Kridli, G. T., Friedman, P. A., & Boileau, J. M. (2021). Manufacturing processes for light alloys. In *Materials, design and manufacturing for lightweight vehicles* (pp. 267-320). Woodhead Publishing.
- [12] Xu, L., Skoularidou, M., Cuesta-Infante, A., & Veeramachaneni, K. (2019). Modeling tabular data using conditional gan. *Advances in neural information processing systems*, 32.
- [13] Raghu, M. & Ganesh Kumar, D. (2024). A conditional tabular generative adversarial network (CTGAN)-based approach to safeguarding artificially created smart IoT setting. *International Innovative Research Journal of Engineering and Technology*. 9(4), 1-9. <https://doi.org/10.32595/iirjet.org/v9i4.2024.194>.
- [14] Topol, E.J. (2019). High-performance medicine: The convergence of human and artificial intelligence. *Nature Medicine*. 25, 44-56. DOI: 10.1038/s41591-018-0300-7
- [15] Rajkomar, A., Oren, E., Chen, K., Dai, A. M., Hajaj, N., Hardt, M., Liu, P.J., Liu, X., Marcus, J., Sun, M., Sundberg, P., Yee, H., Zhang, K., Zhang, Y., Zhang, Y., Flores, G., Duggan, G.E., Irvine, J., Le, Q., Litsch, K., Mossin, A., Tansuwan, J., Wang, D., Wexler, J., Wilson, J., Ludwig, D., Volchenboun, S.L., Chou, K., Pearson, M., Madabushi, S., Shah, N.H., Butte, A.J., Howell, M.D., Cui, C., Corrado, G.S. & Dean, J. (2018). Scalable and accurate deep learning with electronic health records. *NPJ Digital Medicine*. 1(1), 18, 1-10. DOI: 10.1038/s41746-018-0029-1
- [16] Bayat, A. (2002). Science, medicine, and the future: Bioinformatics. *BMJ*. 324, 1018-1022. DOI: 10.1136/bmj.324.7344.1018
- [17] Zhu, Y., Qiu, P., Ji, Y. (2014). TCGA-Assembler: Open-source software for retrieving and processing TCGA data. *Nature methods*. 11(6), 599-600. <https://doi.org/10.1038/nmeth.2956>.
- [18] Zhu, Y., Xu, Y., Helseth Jr, D.L., Gulukota, K., Yang, S., Pesce, L.L., Mitra, R., Muller, P., Sengupta, S., Guo, W., Silverstein, J.C., Foster, I., Parsad, N., White, K.P. & Ji, Y. (2015). A comprehensive depiction of genetic interactions in cancer by integrating TCGA data. *Journal of the National Cancer Institute*. 107(8), 129, 1-9. DOI: 10.1093/jnci/djv129.
- [19] Taha, M.E., Mahmoud, T.M. & Abd-El-Hafeez, T. (2023). A novel hybrid approach to masked face recognition using robust PCA and GOA optimizer. *Scientific Journal for Damietta Faculty of Science*. 13(3) 25-35. DOI: 10.21608/SJDFS.2023.222524.1117.
- [20] Elmessery, W.M., Maklakov, D.V., El-Messery, T.M., Baranenko, D.A., Gutiérrez, J., Shams, M.Y., El-Hafeez, T.A., Elsayed, S., Alhag, S.K., Moghanm, F.S., Mulyukin, M.A., Petrova, Y.Y. & Elwakeel, A.E. (2024). Semantic segmentation of microbial alterations based on SegFormer. *Frontiers in Plant Science*. 15, 1352935, 1-20. DOI: 10.3389/fpls.2024.1352935.
- [21] Girgis, M.R., Mahmoud, T.M., & Abd-El-Hafeez, T. (2010). A new effective system for filtering pornography images from web pages and PDF files. *International Journal of Web Applications*. 2(1), 1-13.
- [22] Eman, M., Mahmoud, T.M., Ibrahim, M.M. & Abd El-Hafeez, T. (2023). Innovative hybrid approach for masked face recognition using pretrained mask detection and segmentation, robust PCA, and KNN classifier. *Sensors*. 23(15), 6727, 1-20. DOI:10.3390/s23156727.
- [23] Abd El-Hafeez, T. (2010). A new system for extracting and detecting skin color regions from PDF documents. *International Journal on Computer Science and Engineering*. 9(2), 2838-2846. ISSN : 0975-3397.
- [24] Mahmoud, T., Abd El-Hafeez, T. & Omar, A. (2014). A highly efficient content based approach to filter pornography websites. *International Journal of Computer Vision and Image Processing*. 2(1), 75-90. DOI:10.4018/ijcvip.2012010105.
- [25] Ali, A., Abd El-Hafeez, T. & Mohany, Y. (2019). A robust and efficient system to detect human faces based on facial features. *Asian Journal of Research in Computer Science*. 2(4), 1-12. DOI:10.9734/AJRCOS/2018/v2i430080.
- [26] Saabia, A.A.B., El-Hafeez, T., Zaki, A.M. (2019). Face recognition based on grey wolf optimization for feature selection. In: Hassanien, A., Tolba, M., Shaalan, K., Azar, A. (Eds.), *Proceedings of the International Conference on*

- Advanced Intelligent Systems and Informatics 2018*. Springer, Cham. DOI: 10.1007/978-3-319-99010-1\_25.
- [27] Ali, A.A., El-Hafeez, T.A. & Mohany, Y.K. (2019). An accurate system for face detection and recognition. *Journal of Advances in Mathematics and Computer Science*. 33(3), 1-19. DOI:10.9734/jamcs/2019/v33i330178.
- [28] Sharma, A., Vans, E., Shigemizu, D., Boroevich, K.A. & Tsunoda, T. (2019). DeepInsight: A methodology to transform a non-image data to an image for convolution neural network architecture. *Scientific Reports*. 9, 11399, 1-7. DOI: 10.1038/s41598-019-47765-6.
- [29] Bazgir, O., Zhang, R., Dhruba, S. R., Rahman, R., Ghosh, S. & Pal, R. (2020) Representation of features as images with neighborhood dependencies for compatibility with convolutional neural networks. *Nature Communications*. 11, 4391, 1-13. DOI: 10.1038/s41467-020-18197-y.
- [30] Ma, S., Zhang, Z. (2018). OmicsMapNet: Transforming omics data to take advantage of deep convolutional neural network for discovery. *Machine Learning*. 1, 1-33. <https://arxiv.org/abs/1804.05283>.
- [31] Van der Maaten, L.J.P., Hinton, G.E. (2008). Visualizing high-dimensional data using t-SNE. *Journal of Machine Learning Research*. 9(11), 2579-2605.
- [32] Nikhil, S., Sah, R.K., Parki, S.K., Tamang, T.B. & TR, M. (2023) Stock market prediction using genetic algorithm assisted LSTM-CNN hybrid model. In 2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT), 6-8 July 2023 (pp. 1-6). Delhi, India. DOI: 10.1109/ICCCNT56998.2023.10306948.
- [33] Shneiderman, B. (1992). Tree visualization with tree-maps: 2-d space-filling approach. *ACM Transactions on Graphics*. 11(1), 92-99. <https://doi.org/10.1145/102377.115768>.
- [34] Zhu, Y., Brettin, T., Xia, F. & Partin, A. (2021). Converting tabular data into images for deep learning with convolutional neural networks. *Scientific Reports*. 11(1), 11325, 1-11. DOI:10.1038/s41598-021-90923-y.
- [35] Okuniewska, A., Perzyk, M. & Kozłowski, J. (2021). Methodology for diagnosing the causes of die-casting defects, based on advanced big data modelling. *Archives of Foundry Engineering*. 21(4), 103-109. DOI:10.24425/afe.2021.138687.
- [36] Okuniewska, A. (2023) *Methodology for diagnosing the causes of product defects on the basis of advanced modelling based on big data sets*. Retrieved December 13, 2024, from <https://www.bip.pw.edu.pl/Postepowania-w-sprawie-nadania-stopnia-naukowego/Doktoraty/Wszczete-po-30-kwietnia-2019-r/Rada-Naukowa-Dyscypliny-Inzynieria-Mechaniczna/mgr-inz.-Alicja-Okuniewska/rozprawa-doktorska>.
- [37] Okuniewska, A., Perzyk, M. & Kozłowski, J. (2023). Machine learning methods for diagnosing the causes of die-casting defects. *Computer Methods in Materials Science*. 23(2), 45-56. DOI:10.7494/cmms.2023.2.0809.
- [38] SDV (2024). Retrieved December 13, 2024, from Available: [https://sdv.dev/SDV/api\\_reference/tabular/api/sdv.tabular.ctgan.CTGAN.html](https://sdv.dev/SDV/api_reference/tabular/api/sdv.tabular.ctgan.CTGAN.html).
- [39] Marin, J. (2022). Evaluating synthetically generated data from small sample sizes: an experimental study. *Machine Learning Computer Science*. 1, 1-24. DOI: 10.48550/arXiv.2211.10760.
- [40] Snoko, J., Raab, G.M., Nowok, B., Dibben, C. & Slavkovic, A. (2018) General and specific utility measures for synthetic data. *Journal of the Royal Society Series, A: Statistics in Society*. 181(3), 663-688. DOI: /10.1111/rssa.12358.
- [41] Carr, A. (2024). *Evaluating data sampling methods with a synthetic quality score*. Retrieved December 13, 2024, from <https://gretel.ai/blog/evaluating-data-sampling-methods-with-a-synthetic-quality-score>.
- [42] Borisov, V., Leemann, T., Seßler, K., Haug, J., Pawelczyk, M., Kasneci, G. (2022). Deep neural networks and tabular data: A survey. *IEEE Transactions on Neural Networks and Learning Systems*. 35(6), 7499-7519. DOI: 10.1109/TNNLS.2022.3229161.
- [43] Fayaz, S.A., Zaman, M., Kaul, S. & Butt, M.A. (2022). Is deep learning on tabular data enough? *An Assessment. International Journal of Advanced Computer Science and Applications*. 13(4), 466-473. DOI: 10.14569/IJACSA.2022.0130454.
- [44] Neto, L., et al., (2023). A comparative analysis of converters of tabular data into image for the classification of Arboviruses using Convolutional Neural Networks. *PLoS One*. 18(12), e0295598, 1-14. DOI:10.1371/journal.pone.0295598.
- [45] Bhatt, D., Patel, C., Talsania, H., Patel, J., Vaghela, R., Pandya, S., Modi, K. & Ghayvat, H. (2021). CNN variants for computer vision: history, architecture, application, challenges and future scope. *Electronics*. 10(20), 2470, 1-28. <https://doi.org/10.3390/electronics10202470>.
- [46] Bouvrie, J (2006). *Introduction Notes on Convolutional Neural Networks*. Retrieved December 13, 2024, from [https://web.mit.edu/jvb/www/papers/cnn\\_tutorial.pdf](https://web.mit.edu/jvb/www/papers/cnn_tutorial.pdf).
- [47] Nwankpa, C., Ijomah, W., Gachagan, A. & Marshall, S. (2018). Activation functions: Comparison of trends in practice and research for deep learning. *Machine Learning ArXiv preprint*. 1-20. <https://doi.org/10.48550/arXiv.1811.03378>.
- [48] Kingma, D.P., Ba, J.A. (2014). A method for stochastic optimization. In 3rd International Conference for Learning Representations. San Diego.
- [49] Rajawat, A.S., Ghosh, A. (2022). Chapter six - Renewable energy system for industrial internet of things model using fusion-AI. In *Applications of AI and IOT in Renewable Energy*. 107-128. DOI: /10.1016/B978-0-323-91699-8.00006-1.
- [50] Kandel, I. & Castelli, M. (2020). The effect of batch size on the generalizability of the convolutional neural networks on a histopathology dataset. *ICT Express*. 6(4), 312-315. DOI: 10.1016/j.icte.2020.04.010.
- [51] Burzyńska, A. (2024). Review of data-driven decision support systems and methodologies for the diagnosis of casting defects. *Archives of Foundry Engineering*. 24(4), 126-135. DOI: 10.24425/afe.2024.151320.