Article

Not peer-reviewed version

An Anomaly Detection Approach to Determine Optimal Cutting Time in Cheese Formation

Andrea Loddo * , Davide Ghiani , Alessandra Perniciano , Luca Zedda , Barbara Pes , Cecilia Di Ruberto

Posted Date: 29 April 2024

doi: 10.20944/preprints202404.1882.v1

Keywords: image processing; computer vision; machine learning; food industry; curd-firming time detection

Article

An Anomaly Detection Approach to Determine Optimal Cutting Time in Cheese Formation

Andrea Loddo[†], **Davide Ghiani** [†] **Alessandra Perniciano** [†] **Luca Zedda** [†] **Barbara Pes** [†] **and Cecilia Di Ruberto** [*,†]

Department of Mathematics and Computer Science, University of Cagliari, Via Ospedale 72, 09124, Cagliari, Italy
[*] Correspondence: andrea.loddo@unica.it; (A.L.)
[†] These authors contributed equally to this work.

Abstract: The production of cheese, a beloved culinary delight worldwide, faces challenges in maintaining consistent product quality and operational efficiency. One crucial stage in this process is determining the precise cutting time during curd formation, which significantly impacts the quality of the cheese. Misjudging this timing can lead to the production of inferior products, harming a company's reputation and revenue. Conventional methods often fall short of accurately assessing variations in coagulation conditions due to the inherent potential for human error. To address this issue, we propose an anomaly detection-based approach. In this approach, we treat the class representing curd formation as the anomaly to be identified. Our proposed solution involves utilizing a one-class, fully convolutional data description network, which we compared against several state-of-the-art methods to detect deviations from the standard coagulation patterns. Encouragingly, our results show F1 scores of up to 0.92, indicating the effectiveness of our approach.

Keywords: image processing; computer vision; machine learning; food industry; curd-firming time detection

1. Introduction

Dairy products possess intrinsic qualities that enhance gastrointestinal tract health and contribute to the well-being of the human microbiome. They play a pivotal role in the food industry due to their rich content of protein, calcium, and micronutrients, all of which are crucial for maintaining bone and muscle health. Among dairy products, cheese stands out as one of the most widely consumed and versatile options globally. With its diverse array of flavors and forms, cheese holds a prominent position in culinary culture, contributing significantly to dietary diversity and enjoyment.

Cheese is a fundamental ingredient in numerous culinary recipes and is often enjoyed on its own. Consequently, evaluating its quality becomes essential for consumers and the industry [1].

In the cheese manufacturing process, curd represents a crucial intermediate stage achieved by heating milk and introducing rennet. Rennet induces the coagulation of casein granules in the milk, resulting in the formation of curd, which settles at the bottom, accompanied by the generation of whey. However, in many cheese varieties, the natural separation of whey and curd does not occur spontaneously, necessitating the mechanical cutting of the coagulated mass into small cubes, referred to as curd grains [2].

As demonstrated by Johnson et al. [3], the coagulation process induced by rennet during cheese production, and consequently the timing of curd cutting, significantly impacts cheese quality.

Determining the cutting time is contingent upon the rheological and microstructural properties of the curd gels, which are influenced by various factors, including milk pretreatment, composition, and coagulation conditions. Consequently, the identification of the cutting time, manually performed by a diary operator, varies across different cheese varieties and profoundly affects parameters such as moisture content, yield, and overall quality of the cheese, as well as losses in whey fat [2].

This challenge is notably accentuated in large-scale automated production facilities, where the variability in coagulation conditions, process alterations, and the potential for human errors introduce complexities in maintaining precise control over cutting times [2,4,5].

For these reasons, integrating advanced methodologies, such as computer vision (CV) techniques, becomes indispensable to mitigate the issues, improve the cheese-making process, enhance production efficiency, and optimize product quality.

In light of the challenging task posed by identifying the optimal cutting time in cheese production, we approach it through the lens of anomaly detection. Given the abundance of images depicting the normal condition of the milk before its cutting time, we adopt an anomaly detection setup to discern anomalies within this dataset. We consider the curd spot an anomaly, seeking to identify it amidst the distribution of normal curd images. By leveraging this approach, we aim to effectively identify deviations from the standard milk appearance, thereby facilitating the accurate determination of the curd and related cutting time.

Specifically, we propose adapting a deep learning technique belonging to the realm of one-class classification, termed Fully Convolutional Data Description (FCDD). This method employs a neural network to reconfigure the data such that normal instances are centered around a predefined focal point while anomalous instances are situated elsewhere. Additionally, a sampling technique transforms the data into images representing a heatmap of subsampled anomalies. Pixels in this heatmap distant from the center correspond to anomalous regions within the input image. FCDD exclusively utilizes convolutional and pooling layers, thereby constraining the receptive field of each output pixel [6]. Moreover, we also compared our findings with more classical machine learning (ML) approaches, specifically trained with handcrafted (HC) and deep features. The latter were extracted by pre-trained Convolutional Neural Network (CNN) architectures.

The rest of the manuscript is organized as follows. Section 2 provides a comprehensive review of existing methodologies for analyzing milk coagulation and anomaly detection techniques. Section 3 elucidates details regarding the dataset, feature extraction methodologies, classifiers adopted, and evaluation measures. Section 4 delves into the experimental evaluation conducted, offering a presentation of the undertaken experiments along with the corresponding results and subsequent discussions. The concluding remarks of this study, along with insightful suggestions for potential enhancements and avenues for future research based on our findings, are given in Section 5.

2. Related Work

This section gives an overview of the existing automated methods in the dairy industry (Section 2.1) and the anomaly detection techniques (Section 2.2).

2.1. *Automated Methods in Dairy Industry*

The analysis of milk-related products often demands specialized acquisition techniques such as fluorescence spectroscopy, an effective spectral approach for determining the intensity of fluorescent components in cheese, and near-infrared spectroscopy, aimed at developing non-destructive assessment methods [7,8].

CV methods necessitate the utilization of cameras and controlled illumination setups to capture images of dairy products. These techniques exhibit versatility, extending beyond the assessment of optimal cutting times [9]. Combining CV methods with artificial intelligence (AI) has found application in various tasks, including the classification of cheese ripeness in entire cheese wheels [10] and the inspection and grading of cheese meltability [11].

Moreover, beyond production processes, the integration of AI in the analysis of dairy products extends to estimating the shelf life of such products [12,13].

However, recent advancements, driven by the scarcity of dairy-related data, aim to tackle challenging tasks such as detecting adulteration or identifying rare product-compromising events through the utilization of anomaly detection approaches [14,15].

2.2. Anomaly Detection

Anomaly detection, the identification of patterns or instances that do not conform to expected behavior, has garnered significant attention across various domains due to its critical importance in identifying potential threats, faults, or outliers, with several key methodologies and approaches [6,16,17].

Statistical methods form the foundation of many anomaly detection techniques. One of the earliest approaches is based on statistical properties such as mean, variance, and probability distributions. Techniques like Z-score, Grubbs', and Dixon's Q-test utilize statistical thresholds to identify outliers [18–20]. However, these methods often assume normality and may not effectively capture complex patterns in high-dimensional data.

ML techniques have gained prominence in anomaly detection due to their ability to handle complex data patterns. Two examples of ML algorithm modifications for anomaly detection are One-Class SVM (OCSVM)[21] and Isolation Forest (IF)[22], which are the adaptation of Support Vector Machines (SVM) and Random Forest (RF), respectively.

In this context, ensemble methods combine multiple anomaly detection algorithms to improve overall performance and robustness. Techniques like IF and Local Outlier Factor leverage ensemble principles to identify anomalies by aggregating results from multiple base learners. Hybrid approaches integrate different anomaly detection techniques, leveraging the strengths of each method to enhance detection accuracy and reliability [23].

Clustering-based methods (e.g., k-means, DBSCAN) and density estimation techniques (e.g., Gaussian Mixture Models) are widely used for anomaly detection without requiring labeled data. These methods identify outliers based on deviations from normal data distributions. However, they may struggle with high-dimensional or sparse data and are sensitive to parameter settings [24,25].

Deep learning (DL) techniques, particularly neural networks, have shown promising results in anomaly detection tasks. Autoencoder-based architectures, such as Variational Autoencoders (VAEs) [26] and Generative Adversarial Networks (GANs) [27], learn compact representations of normal data and detect anomalies based on reconstruction errors. Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) networks are effective in capturing temporal dependencies, making them suitable for sequential data anomaly detection tasks [28,29].

3. Materials and Methods

3.1. Dataset

The dataset was assembled by collecting images from a dairy company based in Sardinia, Italy. The image acquisition process involved using a Nikon D750 camera equipped with a CMOS sensor measuring 35.9×24.0 mm and a resolution of 24 megapixels. All images are in RGB format, with resolutions of $6,016 \times 4,016$ pixels.

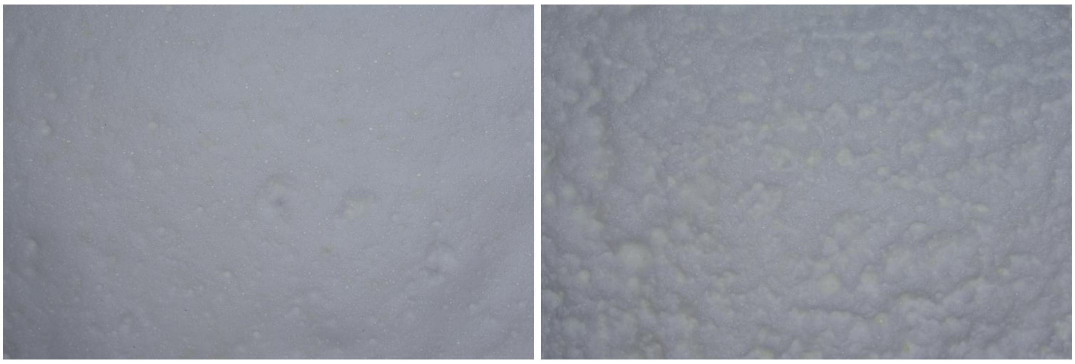
It consists of 12 distinct sets of images. Each one documents the coagulation process as milk transforms from its initial liquid state to the curd stage. More specifically, every set contains two distinct classes: one with images representing the normal curd condition, identified as **non-target** (i.e., the normal instances), and one with images representing the optimal moments for cutting time, identified as **target** (i.e., the anomaly instances).

Table 1 provides a comprehensive summary of each set, numbered from 1 to 12, including details such as the number of images and the class composition. As can be seen, the dataset has a significant class imbalance, with most images falling into the **non-target** class. Sets 1 and 2 depict the coagulation process of fresh whole sheep's milk (*Pecorino Romano*), while subsequent sets involve a blend of cow and sheep milk.

Figure 1 presents two sample images from Set 11, visualizing the differences between the two classes.

Table 1. Comprehensive dataset details: the table provides information on the 12 distinct sets of time-ordered images, denoted from 1 to 12. Each set is characterized by its number of images, representing the number of images it contains, and their distribution between the two classes.

Set	1	2	3	4	5	6	7	8	9	10	11	12
Number of images	94	102	128	112	77	70	84	96	105	94	89	111
Non-target samples	77	90	108	89	54	45	63	72	68	59	60	77
Target samples	17	12	20	23	23	25	21	24	37	35	29	34



Negative (non-target) class

Positive (target) class

Figure 1. Sample images from Set 11: on the left, a picture representing the negative (non-target) class; on the right, a picture representing the positive (target) class.

3.2. Feature Extraction

This section summarizes the feature extraction process adopted to train the ML methods used in this study.

3.2.1. Handcrafted Features

HC features encompass diverse techniques and methodologies to extract morphological, pixel-level, and textural information from images. These features can be categorized into three primary categories: invariant moments, textural features, and color-based features [30]. Each category is briefly described below, while every parameter has been set by considering approaches in similar contexts [10].

Invariant Moments. Denoted as the weighted averages of pixel intensities within an image, they are used to extract specific image properties, aiding in characterizing segmented objects. In this study, three distinct types of moments—Zernike, Legendre, and Chebyshev—were used.

Chebyshev Moments (CH). Introduced by Mukundan and Ramakrishnan [31] and derived from Chebyshev polynomials, they were employed with both first-order (CH_1) and second-order (CH_2) moments of order 5.

Legendre Moments (LM). Initially proposed by Teague [32] and derived from Legendre orthogonal polynomials [33], they were used with second-order of order 5.

Zernike Moments (ZM). Introduced by Oujaoura et al. [34] and derived from Zernike polynomials, they were applied with order 6 and a repetition of 4.

Texture features. They focus on fine textures with different approaches. Here, the following were used.

Haar Features (Haar). Consisting of adjacent rectangles with alternating positive and negative polarities, they were used in various forms, such as edge features, line features, four-rectangle features, and center-surround features. Haar features play a crucial role in cascade classifiers as part of the Viola-Jones object detection framework [35].

Rotation-Invariant Haralick Features (HARri). Thirteen Haralick features [36], derived from the Gray Level Co-occurrence Matrix (GLCM), were transformed into rotation-invariant features [37].

This transformation involved computing GLCM variations with parameters set to $d = 1$ and angular orientations $\theta = [0°, 45°, 90°, 135°]$.

Local Binary Pattern (LBP). As described by Ojala et al. [38], LBP characterizes texture and patterns within images. In this work, the histogram of the LBP, converted to a rotation-invariant form (LBP_ri) [39], was extracted using a neighborhood defined by a radius $r = 1$ and a number of neighbors $n = 8$.

Color features. They aim to extract color intensity information from images. In this study, these descriptors were calculated from images converted to grayscale, simplifying the analysis and computation process.

Grayscale Histogram Features (Hist): The color histogram characterizes the global color distribution within images. Seven statistical descriptors, including mean, standard deviation, smoothness, skewness, kurtosis, uniformity, and entropy, were computed from the grayscale histogram.

3.3. Deep Features

CNNs have demonstrated their effectiveness as deep feature extractors in different contexts [40–42] as well as in anomaly detection setups [43–45]. CNNs excel at capturing global features from images by processing the input through multiple convolutional filters and progressively reducing dimensionality across various architectural stages. For our experiments, we selected several architectures pre-trained on the Imagenet1k dataset [46], as presented in Table 2, along with comprehensive details regarding the selected layers for feature extraction, input size, and the number of trainable parameters for each architecture.

Table 2. Employed convolutional neural networks details including reference paper, number of trainable parameters in millions, input shape, feature extraction layer and related feature vector size.

Ref.	Params (M)	Input shape	Feature layer	# Features
AlexNet [47]	60	224 × 224	Pen. FC	4,096
DarkNet-53 [48]	20.8	224 × 224	Conv53	1,000
DenseNet-201 [49]	25.6	224 × 224	Avg. Pool	1,920
GoogLeNet [50]	5	224 × 224	Loss3	1,000
EfficientNetB0 [51]	5.3	224 × 224	Avg. Pool	1,280
Inception-v3 [52]	21.8	299 × 299	Last FC	1,000
Inception-ResNet-v2 [53]	55	299 × 299	Avg. pool	1,536
NasNetL [54]	88.9	331 × 331	Avg. Pool	4,032
ResNet-18 [55]	11.7	224 × 224	Pool5	512
ResNet-50 [55]	26	224 × 224	Avg. Pool	1,024
ResNet-101 [55]	44.6	224 × 224	Pool5	1,024
VGG16 [56]	138	224 × 224	Pen. FC	4,096
VGG19 [56]	144	224 × 224	Pen. FC	4,096
XceptionNet [57]	22.9	299 × 299	Avg. Pool	2,048

CNNs have exhibited their efficacy as feature extractors across diverse domains [40–42], including anomaly detection setups [43–45]. CNNs excel at capturing global features from images by subjecting the input data to multiple convolutional filters and progressively reducing dimensionality across different architectural stages. In our experimental setup, we opted for several pre-trained architectures on the Imagenet1k dataset [46], as delineated in Table 2. Comprehensive details pertaining to the selected layers for feature extraction, input size, and the number of trainable parameters for each architecture are provided.

3.4. ML and DL methods

This section presents the classification methods employed in our analysis. As suggested in Section 2, we selected three classifiers that could cope with the data to analyze, taking into account that each set is unique and potentially requires different methods for analysis. Also, we aimed to compare three baseline methods to obtain insights into a novel data set with limited images. For this reason, we avoid using more complex DL-based methods, such as VAEs or GANs, as we face a relatively small sample size dataset.

More specifically, we selected two ML-based classifiers: the OCSVM and the IF. Finally, we chose the FCCD network from the DL-based approaches. The former two were trained with every single feature described in Section 3.2 while being a DL approach, the latter was trained end-to-end. A brief description of these methods is now given.

3.4.1. One-Class SVM

As introduced in Section 2, this ML algorithm belongs to the family of SVMs and is primarily designed for anomaly detection in datasets where only one class, typically the majority class (normal instances), is available for training.

It aims to learn a decision boundary that encapsulates the normal instances in the feature space. This boundary is constructed in such a way that it maximizes the margin between the normal instances and the hyperplane while minimizing the number of instances classified as outliers. Unlike traditional SVMs, which aim to find a decision boundary that separates different classes, the One-Class SVM focuses solely on delineating the region of normality [58].

After training, the One-Class SVM can classify new instances as either normal or anomalous based on their distance from the decision boundary. Instances within the margin defined by the support vectors are considered normal, while those outside the margin are classified as anomalies.

3.4.2. Isolation Forest

It is an ensemble-based anomaly detection algorithm that operates on the principle of isolating anomalies rather than modeling normal data points. It is particularly effective in identifying anomalies in high-dimensional datasets and is capable of handling both numerical and categorical features [59].

The main idea behind the Isolation Forest algorithm is to isolate anomalies by constructing random decision trees. Unlike traditional decision trees that aim to partition the feature space into regions containing predominantly one class, Isolation Forest builds trees that partition the data randomly. Specifically, each tree is constructed by recursively selecting random features and splitting the data until all instances are isolated in leaf nodes.

The anomaly score assigned to each instance is based on the average path length in the trees. Anomalies are expected to have shorter average path lengths compared to normal instances, making them distinguishable from the majority of data points. The anomaly score can be thresholded to identify outliers, with instances exceeding the threshold considered anomalies.

3.4.3. FCDD Network

It is a DL technique originally proposed by Liznerski et al. [6] for anomaly detection, particularly within the framework of one-class classification. FCDD employs a backbone CNN composed solely of convolutional and pooling layers devoid of fully connected layers.

At its core, FCDD aims to transform the input data such that normal instances are concentrated around a predetermined center while anomalous instances are situated elsewhere in the feature space. This transformation enables the model to effectively distinguish between normal and anomalous patterns in the data distribution.

One distinctive feature of FCDD is its utilization of a sampling method to convert data samples into images representing heatmaps of subsampled anomalies. These heatmaps visualize the anomalies present in the input data, with pixels farther from the center corresponding to anomalous regions within the original images. An example is shown in Figure 2.

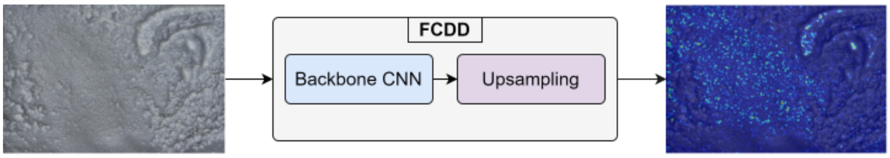


Figure 2. Schematic representation of generating full-resolution anomaly heatmaps using FCDD approach. On the left, the original input image is taken in input by the backbone CNN. An upsampling procedure is then applied to the scaled CNN's output, by means of a transposed Gaussian convolution, to obtain a full-size heatmap [6].

By employing convolutional and pooling layers exclusively, FCDD restricts the receptive field of each output pixel, allowing for localized feature extraction and preserving spatial information within the data. This characteristic makes FCDD particularly suitable for processing image data, where spatial relationships and local patterns play a crucial role in anomaly detection.

3.5. Evaluation Measures

Consider a binary classification task where each example e is represented by a pair $\langle i, t \rangle$, with i denoting feature values and t representing the target category. A dataset D comprises such examples. In this binary scenario, the categories are typically labeled as *negative* and *positive*.

Performance evaluation of a binary classifier on dataset D involves labeling each instance as *negative* or *positive* based on the classifier's output. The evaluation is based on the following metrics:

- *True negatives (TN)*: Instances correctly predicted as *negative*.
- *False positives (FP)*: Instances incorrectly predicted as *positive*.
- *False negatives (FN)*: Instances incorrectly predicted as *negative*.
- *True positives (TP)*: Instances correctly predicted as *positive*.

These metrics lead to the following definitions:

- *Precision (P)*: The fraction of positive instances correctly classified among all instances classified as positive:
$$P = \frac{TP}{TP + FP} \tag{1}$$
- *Recall (R)* (or *sensitivity*): Measures the classifier's ability to predict the positive class against FN (also known as true positive rate):
$$R = \frac{TP}{TP + FN} \tag{2}$$
- *F1-score (F1)*: The harmonic mean between precision and recall:
$$F1 = 2 \cdot \frac{P \cdot R}{P + R} \tag{3}$$

4. Experimental Results

In this section, we comprehensively explore the interpretation and implications of the results derived from our study. We structure our analysis into three distinct sections: firstly, we present the outcomes obtained using various shallow learning classifiers with handcrafted features and deep learning-based features. Following this, we discuss the results achieved by employing an FCDD network as an anomaly detection method. However, to simplify our discussion, we report only the best-performing pairs of classifiers and features. This systematic approach provides a detailed examination of the efficacy and nuances of each method employed, highlighting valuable insights into their respective performance. Finally, we provide a global experiment result analysis.

4.1. Experimental Setup

In this research, all the selected classifiers were trained with default parameters to prevent the generalization capability of the trained models from being influenced. For the sake of brevity, we reported only the results obtained with the best ML classifier based on the F1-score value obtained on the Target class, which is the OCSVM. We also provide the outcomes achieved through the FCDD method.

The experiments conducted with ML techniques have been carried out by ranging over eight HC feature categories and fourteen deep features extracted from off-the-shelf, pre-trained CNNs. Both categories are described in Section 3.2. Moreover, the classifiers were trained on the Non-Target class only, following the one-class classification paradigm [17].

The evaluation strategy was 5-fold cross-validation. The FCDD network was composed of a ResNet-18 as the backbone. It was trained with the following splits: 50%, 10%, 40% for training, calibration, and test for the normal (non-target) class.Each training was repeated 5 times. Moreover, the training procedures were executed for a maximum of 100 epochs, with early stopping based on the calibration set performance. A batch size of 32, the Adam optimizer, and a learning rate of 0.001 were employed.

Relevant performance measures have been reported for each experiment. In particular, for both classes, we reported precision, recall, and F1, as described in Section 3.5.

Experiments were executed on a workstation equipped with the following hardware: an Intel(R) Core(TM) i9-8950HK @ 2.90GHz CPU, 32 GB RAM, and an NVIDIA GTX1050 Ti GPU with 4GB of memory.

4.2. Quantitative Results

As previously said, here we present the results obtained with the best ML-based approach, OCSVM, when trained with HC or deep features and an FCDD network.

4.2.1. Results with ML Approaches and HC Features

The results of this approach are summarized in Table 3. In this setting, OCSVM consistently outperformed IF on every single set. Significant differences are observed in F1 values for the Target class, ranging from 0.72% (set 12) to 0.91 (set 9), indicating how some sets are more challenging than others. Even though the precision for the Target class is not particularly high (constantly below 0.80% except for set 9), the recall scores basically confirm the discrete results on this class.

Table 3. Results obtained with the best ML-based approaches (One-Class SVM) trained on each set with HC features. Average performance over the k folds ($k = 5$) and standard deviation (the latter within round brackets) for each set and class are reported. The feature column reports the best HC feature for the specific set.

		Non-Target			Target		
Set	Features	Precision	Recall	F1	Precision	Recall	F1
1	ZM	0.97 (0.07)	0.61 (0.08)	0.74 (0.08)	0.75 (0.02)	0.99 (0.01)	0.85 (0.02)
2	ZM	0.94 (0.03)	0.62 (0.03)	0.75 (0.03)	0.62 (0.03)	0.93 (0.02)	0.75 (0.02)
3	CH_2	0.81 (0.11)	0.76 (0.09)	0.78 (0.10)	0.75 (0.02)	0.80 (0.02)	0.77 (0.02)
4	ZM	0.88 (0.07)	0.48 (0.27)	0.62 (0.14)	0.71 (0.06)	0.95 (0.02)	0.81 (0.09)
5	Haar	1.00 (0.05)	0.11 (0.35)	0.2 (0.15)	0.71 (0.06)	1.00 (0.01)	0.83 (0.03)
6	ZM	0.40 (0.03)	0.16 (0.03)	0.21 (0.03)	0.76 (0.03)	0.94 (0.03)	0.84 (0.04)
7	ZM	1.00 (0.03)	0.38 (0.03)	0.52 (0.03)	0.73 (0.04)	1.00 (0.01)	0.84 (0.03)
8	ZM	1.00 (0.02)	0.25 (0.03)	0.40 (0.03)	0.69 (0.14)	1.00 (0.01)	0.82 (0.03)
9	Hist	0.82 (0.03)	0.70 (0.04)	0.75 (0.03)	0.90 (0.00)	0.93 (0.01)	0.91 (0.01)
10	ZM	0.69 (0.14)	0.22 (0.07)	0.30 (0.09)	0.79 (0.04)	0.98 (0.01)	0.87 (0.02)
11	Hist	1.00 (0.04)	0.30 (0.14)	0.45 (0.11)	0.78 (0.04)	1.00 (0.01)	0.87 (0.02)
12	ZM	0.40 (0.03)	0.39 (0.03)	0.39 (0.03)	0.72 (0.02)	0.72 (0.02)	0.72 (0.02)

However, the main issue with this setting lies in the classification of the Non-Target class, which can determine the identification of a wrong cutting time and, therefore, a sub-optimal product. Despite some high precision scores (e.g., 1.00 on set 5, 7, 8, and 11), the recall values demonstrated several misclassification, since no one surpasses 0.76.

From a feature point of view, the invariant moments, ZM in particular, resulted in the best in 8 out of 12 sets, demonstrating superior performance compared to the other HC features.

4.2.2. Results with ML Approaches and Deep Features

The summarized results are presented in Table 4. In this setting, OCSVM outperformed IF on 9 out of 12 sets. In fact, we acknowledge that IF obtained scores between 0.98 and 0.99 for both classes on sets 7, 8, and 9. However, OCSVM obtained a higher F1 score on average in the Target class. For this reason, we report the results obtained with OCSVM.

Table 4. Results obtained with the best ML-based approaches (One-Class SVM) trained on each set with deep features. Average performance over the k folds ($k = 5$) and standard deviation (the latter within round brackets) for each set and class are reported. The feature column reports the best deep feature for the specific set.

		Non-Target			Target		
Set	Features	Precision	Recall	F1	Precision	Recall	F1
1	XceptionNet	1.00 (0.01)	0.47 (0.07)	0.64 (0.04)	0.67 (0.02)	1.00 (0.01)	0.81 (0.01)
2	ResNet-18	1.00 (0.01)	0.41 (0.11)	0.57 (0.09)	0.54 (0.04)	1.00 (0.01)	0.70 (0.05)
3	EfficientNetB0	1.00 (0.01)	0.56 (0.07)	0.71 (0.08)	0.68 (0.03)	1.00 (0.01)	0.81 (0.01)
4	EfficientNetB0	1.00 (0.01)	0.54 (0.04)	0.68 (0.06)	0.75 (0.05)	1.00 (0.01)	0.85 (0.01)
5	XceptionNet	0.79 (0.02)	0.68 (0.03)	0.72 (0.02)	0.86 (0.01)	0.90 (0.02)	0.87 (0.02)
6	EfficientNetB0	0.80 (0.01)	0.31 (0.14)	0.44 (0.12)	0.80 (0.02)	1.00 (0.01)	0.89 (0.02)
7	XceptionNet	1.00 (0.01)	0.27 (0.04)	0.41 (0.03)	0.70 (0.02)	1.00 (0.01)	0.82 (0.02)
8	Inception-ResNet-v2	1.00 (0.01)	0.32 (0.09)	0.48 (0.03)	0.71 (0.04)	1.00 (0.01)	0.83 (0.02)
9	XceptionNet	1.00 (0.01)	0.29 (0.02)	0.43 (0.03)	0.80 (0.01)	1.00 (0.01)	0.89 (0.02)
10	XceptionNet	1.00 (0.01)	0.50 (0.00)	0.66 (0.02)	0.86 (0.01)	1.00 (0.01)	0.92 (0.03)
11	XceptionNet	1.00 (0.01)	0.27 (0.02)	0.41 (0.01)	0.77 (0.03)	1.00 (0.01)	0.87 (0.02)
12	XceptionNet	1.00 (0.01)	0.65 (0.03)	0.78 (0.01)	0.87 (0.03)	1.00 (0.01)	0.93 (0.03)

Even in this case, the performance varies across different sets and classes. For the Target class, precision values range from 0.54 to 0.87, with recall scores varying between 0.90 and 1.00. Similarly, for the Non-Target class, precision scores range from 0.79 to 1.00, with corresponding recall scores ranging from 0.27 to 0.68.

Even though the deep features showcase notable performance improvements compared to hand-crafted ones, particularly in the Target class, the disastrous results obtained with recall scores on the Non-Target classes almost confirm the results already obtained with HC features.

Considering the architectures, XceptionNet produces the best features in 7 sets out of 12, followed by EfficientNetB0 with 3.

4.2.3. Results with FCCD Network

The results obtained by the FCDD are presented in Table 5. In general, performance is higher than OCSVM and IF as this approach achieves the best results on both classes, demonstrating its ability to address the issue of low recall shown in the previous approaches. It also significantly improves the result's stability across different sets and demonstrates near-perfect precision scores for each set except for the Target class of set 2. The results show an average F1 score of 0.91 for the Non-Target class and 0.89 for the Target class, indicating an increase of 34% and 5%, respectively, compared to shallow learning approaches using deep features.

Table 5. Results obtained with the DL approach, i.e., the FCDD network. Average performance over the k folds ($k = 5$) and standard deviation (the latter within round brackets) for each set and class are reported.

	Non-Target			Target		
Set	Precision	Recall	F1-score	Precision	Recall	F1-score
1	1.00 (0.00)	0.75 (0.05)	0.86 (0.03)	1.00 (0.00)	0.83 (0.03)	0.96 (0.03)
2	1.00 (0.01)	0.92 (0.01)	0.96 (0.02)	0.75 (0.02)	1.00 (0.00)	0.86 (0.01)
3	1.00 (0.01)	0.89 (0.01)	0.94 (0.02)	1.00 (0.01)	0.80 (0.02)	0.89 (0.01)
4	1.00 (0.00)	0.83 (0.03)	0.96 (0.03)	1.00 (0.00)	0.80 (0.02)	0.89 (0.01)
5	1.00 (0.01)	0.75 (0.02)	0.86 (0.03)	1.00 (0.01)	0.83 (0.03)	0.96 (0.03)
6	1.00 (0.01)	0.80 (0.02)	0.89 (0.02)	1.00 (0.01)	0.80 (0.02)	0.89 (0.01)
7	1.00 (0.01)	0.86 (0.01)	0.92 (0.02)	1.00 (0.01)	0.75 (0.02)	0.86 (0.01)
8	1.00 (0.01)	0.88 (0.03)	0.94 (0.01)	1.00 (0.01)	0.80 (0.02)	0.89 (0.01)
9	1.00 (0.01)	0.88 (0.03)	0.94 (0.01)	1.00 (0.01)	0.86 (0.01)	0.92 (0.01)
10	1.00 (0.01)	0.83 (0.03)	0.96 (0.03)	1.00 (0.01)	0.86 (0.01)	0.92 (0.01)
11	1.00 (0.01)	0.86 (0.01)	0.92 (0.01)	1.00 (0.01)	0.83 (0.03)	0.96 (0.03)
12	1.00 (0.01)	0.88 (0.03)	0.94 (0.02)	1.00 (0.01)	0.86 (0.04)	0.92 (0.01)

The outcomes obtained through the FCDD method are detailed in Table 5. Generally, the performance surpasses that of the OCSVM and IF algorithms, as FCDD achieves superior results for both classes. This underscores its efficacy in addressing the issue of low recall observed in prior methodologies. Additionally, FCDD enhances the stability of results across various sets, exhibiting near-perfect precision scores for each set except for the Target class of set 2. Specifically, the average F1 scores for the Non-Target and Target classes are 0.91 and 0.89, respectively. These findings denote an enhancement of 34% and 5%, respectively, compared to ML approaches trained with deep features.

4.3. Qualitative Results

Figure 3 shows the heatmaps obtained on some sets (4, 5, 10) for both Non-Target and Target classes. In particular, FCDD shows how the Target class exhibits several more relevant features than the Non-Target class. This aspect makes sense since the Target class in the examined dataset represents the optimal cutting time, which is given when the coagulated milk mass becomes composed of the small cubes known as curd grains.

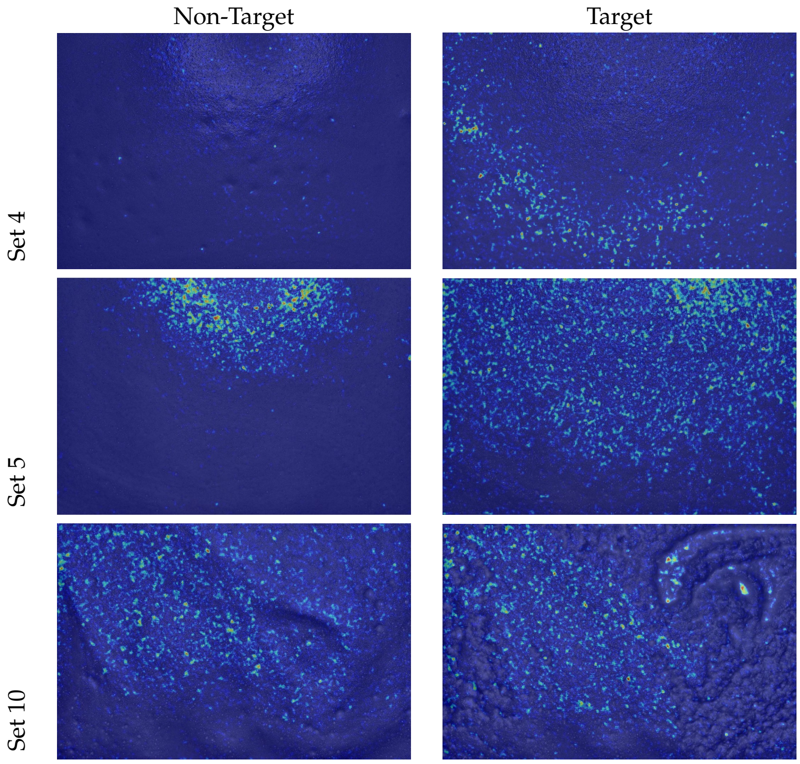


Figure 3. FCDD heatmaps generated for some sample sets obtained on both classes.

5. Conclusions

This study aimed to develop an automated approach for determining cutting time during cheese formation. A key finding is that deep learning-based algorithms, specifically the FCDD network, showed superior performance compared to ML-based classifiers for this task.

The FCDD network achieved an average F1 score of 0.91 and 0.89 for the Non-Target and Target classes, respectively. This demonstrates its ability to accurately identify standard coagulation patterns and deviations, outperforming other evaluated methods. It also showed more stable prediction across different sets.

With ML classifiers, the deep features generally outperformed HC features, and XceptionNet provided the best performance. Furthermore, the OCSVM outperformed the IF in all sets except three in the deep features setting. Despite the key advantage of the IF and OCSVM in their efficiency in handling high-dimensional data, the results have clearly shown that IF struggled with data sets containing structured anomalies, which may be the case. Similarly, OCSVM's performance on the Non-Target class may be sensitive to the choice of kernel function and parameters, such as the nu parameter, which controls the trade-off between the margin size and the number of outliers.

The evaluated FCDD network provides a promising solution for automatically determining optimal cutting time. With the refinement of additional data sets, real-world industrial implementation has the potential to enhance production efficiency and quality control.

Future research may explore and evaluate these approaches based on more extensive and diversified industrial data sets to ascertain the generalizability of the findings. The inclusion of data from a wide array of cheese varieties and production environments has the potential to yield valuable insights. Furthermore, integrating additional modalities, such as thermal imaging, could offer a more comprehensive understanding of the coagulation process, thereby potentially enhancing detection performance within multi-modal frameworks.

To address challenges stemming from limited labeled data availability, the adaptation of self-supervised and semi-supervised learning paradigms may prove advantageous, particularly through the pre-training of models on extensive unlabeled datasets. Moreover, the development of uncertainty estimation techniques, as well as the investigation of further backbones, could facilitate the calibration of predictions based on input ambiguity, thereby enhancing practical deployment in real-world scenarios.

Focusing on efficiency and deployability factors, designing embedded-targeted networks tailored to considerations like model size and latency could streamline on-device implementation for integrated inspection systems. Finally, exploring domain adaptation and transfer learning approaches holds promise for generalizing models in this context.

Author Contributions: Conceptualization, A.L., and C.D.R.; methodology, A.L.; software, A.L., D.G..; validation, A.L., D.G., A.P., L.Z., B.P., C.D.R.; formal analysis, A.L., D.G., A.P., L.Z., B.P., C.D.R.; investigation, A.L.; resources, A.L., D.G.; data curation, A.L.; writing—original draft preparation, A.L., A.P., L.Z., B.P., C.D.R.; writing—review and editing, A.L., A.P., L.Z., B.P., C.D.R.; visualization, A.L., D.G., A.P., L.Z., B.P., C.D.R.; supervision, B.P., C.D.R.; project administration, C.D.R.; funding acquisition, C.D.R.. All authors have read and agreed to the published version of the manuscript.

Funding: We acknowledge financial support under the National Recovery and Resilience Plan (NRRP), Mission 4 Component 2 Investment 1.5 - Call for tender No.3277 published on December 30, 2021 by the Italian Ministry of University and Research (MUR) funded by the European Union – NextGenerationEU. Project Code ECS0000038 – Project Title eINS Ecosystem of Innovation for Next Generation Sardinia – CUP F53C22000430001- Grant Assignment Decree No. 1056 adopted on June 23, 2022 by the Italian Ministry of University and Research (MUR).

Data Availability Statement: We encourage all authors of articles published in MDPI journals to share their research data. In this section, please provide details regarding where data supporting reported results can be found, including links to publicly archived datasets analyzed or generated during the study. Where no new data were created, or where data is unavailable due to privacy or ethical restrictions, a statement is still required. Suggested Data Availability Statements are available in section "MDPI Research Data Policies" at https://www.mdpi.com/ethics.

Data Availability Statement: We employed dataset is publicly available at the following at the following URL.

Acknowledgments: In this section you can acknowledge any support given which is not covered by the author contribution or funding sections. This may include administrative and technical support, or donations in kind (e.g., materials used for experiments).

Conflicts of Interest: Declare conflicts of interest or state "The authors declare no conflicts of interest." Authors must identify and declare any personal circumstances or interest that may be perceived as inappropriately influencing the representation or interpretation of reported research results. Any role of the funders 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 must be declared in this section. If there is no role, please state "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".

Abbreviations

The following abbreviations are used in this manuscript:

MDPI	Multidisciplinary Digital Publishing Institute
DOAJ	Directory of open access journals
TLA	Three letter acronym
LD	Linear dichroism

References

1. Lei, T.; Sun, D.W. Developments of Nondestructive Techniques for Evaluating Quality Attributes of Cheeses: A Review. *Trends in Food Science & Technology* **2019**, *88*. doi:10.1016/j.tifs.2019.04.013.
2. Castillo, M., Cutting Time Prediction Methods in Cheese Making; 2006. doi:10.1081/E-EAFE-120040365.
3. Johnson, M.E.; Chen, C.M.; Jaeggi, J.J. Effect of rennet coagulation time on composition, yield, and quality of reduced-fat cheddar cheese. *J Dairy Sci* **2001**, *84*, 1027–1033.

4. Gao, P.; Zhang, W.; Wei, M.; Chen, B.; Zhu, H.; Xie, N.; Pang, X.; Marie-Laure, F.; Zhang, S.; Lv, J. Analysis of the non-volatile components and volatile compounds of hydrolysates derived from unmatured cheese curd hydrolysis by different enzymes. *LWT* **2022**, *168*, 113896.
5. Guinee, T.P. Effect of high-temperature treatment of milk and whey protein denaturation on the properties of rennet–curd cheese: A review. *International Dairy Journal* **2021**, *121*, 105095.
6. Liznerski, P.; Ruff, L.; Vandermeulen, R.A.; Franks, B.J.; Kloft, M.; Müller, K. Explainable Deep One-Class Classification. 9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021. OpenReview.net, 2021.
7. Alinaghi, M.; Nilsson, D.; Singh, N.; Höjer, A.; Saedén, K.H.; Trygg, J. Near-infrared hyperspectral image analysis for monitoring the cheese-ripening process. *Journal of Dairy Science* **2023**, *106*, 7407–7418. doi:10.3168/jds.2023-23377.
8. Lei, T.; Sun, D.W. Developments of nondestructive techniques for evaluating quality attributes of cheeses: A review. *Trends in Food Science & Technology* **2019**, *88*, 527–542. doi:10.1016/j.tifs.2019.04.013.
9. Everard, C.; O'Callaghan, D.; Fagan, C.; O'Donnell, C.; Castillo, M.; Payne, F. Computer Vision and Color Measurement Techniques for Inline Monitoring of Cheese Curd Syneresis. *90*, 3162–3170.
10. Loddo, A.; Di Ruberto, C.; Armano, G.; Manconi, A. Automatic Monitoring Cheese Ripeness Using Computer Vision and Artificial Intelligence. *IEEE Access* **2022**, *10*, 122612–122626. doi:10.1109/ACCESS.2022.3223710.
11. Badaró, A.T.; De Matos, G.V.; Karaziack, C.B.; Viotto, W.H.; Barbin, D.F. Automated Method for Determination of Cheese Meltability by Computer Vision. *14*, 2630–2641. doi:10.1007/s12161-021-02094-1.
12. Goyal, S.; Goyal, G.K. Shelflife prediction of processed cheese using artificial intelligence ANN technique.
13. Goyal, S.; Goyal, G.K. Smart Artificial Intelligence Computerized Models for Shelf Life Prediction of Processed Cheese. *1*, 281. doi:10.14419/ijet.v1i3.201.
14. Teixeira, J.L.d.P.; Caramês, E.T.d.S.; Baptista, D.P.; Gigante, M.L.; Pallone, J.A.L. Rapid adulteration detection of yogurt and cheese made from goat milk by vibrational spectroscopy and chemometric tools. *96*, 103712. doi:10.1016/j.jfca.2020.103712.
15. Vasafi, P.S.; Paquet-Durand, O.; Brettschneider, K.; Hinrichs, J.; Hitzmann, B. Anomaly detection during milk processing by autoencoder neural network based on near-infrared spectroscopy. *299*, 110510. doi:10.1016/j.jfoodeng.2021.110510.
16. Li, Z.; Zhu, Y.; van Leeuwen, M. A Survey on Explainable Anomaly Detection. *ACM Trans. Knowl. Discov. Data* **2024**, *18*, 23:1–23:54. doi:10.1145/3609333.
17. Ruff, L.; Kauffmann, J.R.; Vandermeulen, R.A.; Montavon, G.; Samek, W.; Kloft, M.; Dietterich, T.G.; Müller, K. A Unifying Review of Deep and Shallow Anomaly Detection. *Proc. IEEE* **2021**, *109*, 756–795. doi:10.1109/JPROC.2021.3052449.
18. Samariya, D.; Aryal, S.; Ting, K.M.; Ma, J. A New Effective and Efficient Measure for Outlying Aspect Mining. Web Information Systems Engineering - WISE 2020 - 21st International Conference, Amsterdam, The Netherlands, October 20-24, 2020, Proceedings, Part II; Huang, Z.; Beek, W.; Wang, H.; Zhou, R.; Zhang, Y., Eds. Springer, 2020, Vol. 12343, *Lecture Notes in Computer Science*, pp. 463–474.
19. Nguyen, V.K.; Renault, É.; Milocco, R.H. Environment Monitoring for Anomaly Detection System Using Smartphones. *Sensors* **2019**, *19*, 3834. doi:10.3390/S19183834.
20. Violettas, G.E.; Simoglou, G.; Petridou, S.G.; Mamatas, L. A Softwarized Intrusion Detection System for the RPL-based Internet of Things networks. *Future Gener. Comput. Syst.* **2021**, *125*, 698–714. doi:10.1016/J.FUTURE.2021.07.013.
21. Schölkopf, B.; Williamson, R.C.; Smola, A.; Shawe-Taylor, J.; Platt, J. Support Vector Method for Novelty Detection. Advances in Neural Information Processing Systems; Solla, S.; Leen, T.; Müller, K., Eds. MIT Press, 1999, Vol. 12.
22. Liu, F.T.; Ting, K.M.; Zhou, Z.H. Isolation Forest. 2008 Eighth IEEE International Conference on Data Mining, 2008, pp. 413–422. doi:10.1109/ICDM.2008.17.
23. Siddiqui, M.A.; Stokes, J.W.; Seifert, C.; Argyle, E.; McCann, R.; Neil, J.; Carroll, J. Detecting Cyber Attacks Using Anomaly Detection with Explanations and Expert Feedback. IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2019, Brighton, United Kingdom, May 12-17, 2019. IEEE, 2019, pp. 2872–2876. doi:10.1109/ICASSP.2019.8683212.
24. Li, J.; Izakian, H.; Pedrycz, W.; Jamal, I. Clustering-based anomaly detection in multivariate time series data. *Appl. Soft Comput.* **2021**, *100*, 106919. doi:10.1016/J.ASOC.2020.106919.

25. Falcão, F.; Zoppi, T.; Silva, C.B.V.; Santos, A.; Fonseca, B.; Ceccarelli, A.; Bondavalli, A. Quantitative comparison of unsupervised anomaly detection algorithms for intrusion detection. Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing, SAC 2019, Limassol, Cyprus, April 8-12, 2019; Hung, C.; Papadopoulos, G.A., Eds. ACM, 2019, pp. 318–327. doi:10.1145/3297280.3297314.
26. Liu, W.; Li, R.; Zheng, M.; Karanam, S.; Wu, Z.; Bhanu, B.; Radke, R.J.; Camps, O. Towards Visually Explaining Variational Autoencoders. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020.
27. Ravanbakhsh, M.; Nabi, M.; Sangineto, E.; Marcenaro, L.; Regazzoni, C.S.; Sebe, N. Abnormal event detection in videos using generative adversarial nets. 2017 IEEE International Conference on Image Processing, ICIP 2017, Beijing, China, September 17-20, 2017. IEEE, 2017, pp. 1577–1581. doi:10.1109/ICIP.2017.8296547.
28. Liu, W.; Luo, W.; Lian, D.; Gao, S. Future Frame Prediction for Anomaly Detection - A New Baseline. 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018. Computer Vision Foundation / IEEE Computer Society, 2018, pp. 6536–6545. doi:10.1109/CVPR.2018.00684.
29. Georgescu, M.I.; Barbalau, A.; Ionescu, R.T.; Khan, F.S.; Popescu, M.; Shah, M. Anomaly Detection in Video via Self-Supervised and Multi-Task Learning. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2021, pp. 12742–12752.
30. Putzu, L.; Loddo, A.; Ruberto, C.D. Invariant Moments, Textural and Deep Features for Diagnostic MR and CT Image Retrieval. Computer Analysis of Images and Patterns: 19th International Conference, CAIP 2021, Virtual Event, September 28–30, 2021, Proceedings, Part I; Springer-Verlag: Berlin, Heidelberg, 2021; pp. 287–297. doi:10.1007/978-3-030-89128-2_28.
31. Mukundan, R.; Ong, S.; Lee, P. Image analysis by Tchebichef moments. *IEEE Transactions on Image Processing* **2001**, *10*, 1357–1364. Conference Name: IEEE Transactions on Image Processing, doi:10.1109/83.941859.
32. Teague, M.R. Image analysis via the general theory of moments∗. *J. Opt. Soc. Am.* **1980**, *70*, 920–930. doi:10.1364/JOSA.70.000920.
33. Teh, C.H.; Chin, R. On image analysis by the methods of moments. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **1988**, *10*, 496–513. Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence, doi:10.1109/34.3913.
34. Oujaoura, M.; Minaoui, B.; fakir, M. Image Annotation by Moments. *Moments and Moment invariants - theory and applications* **2014**, pp. 227–252. doi:10.15579/gcr.vol1.
35. Viola, P.; Jones, M. Rapid object detection using a boosted cascade of simple features. Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001, 2001, Vol. 1, pp. I–I. ISSN: 1063-6919, doi:10.1109/CVPR.2001.990517.
36. Haralick, R.M.; Shanmugam, K.; Dinstein, I. Textural Features for Image Classification. *IEEE Transactions on Systems, Man, and Cybernetics* **1973**, *SMC-3*, 610–621. Conference Name: IEEE Transactions on Systems, Man, and Cybernetics, doi:10.1109/TSMC.1973.4309314.
37. Putzu, L.; Di Ruberto, C. Rotation Invariant Co-occurrence Matrix Features. Image Analysis and Processing - ICIAP 2017; Battiato, S.; Gallo, G.; Schettini, R.; Stanco, F., Eds.; Springer International Publishing: Cham, 2017; Lecture Notes in Computer Science, pp. 391–401. doi:10.1007/978-3-319-68560-1_35.
38. He, D.c.; Wang, L. Texture Unit, Texture Spectrum, And Texture Analysis. *IEEE Transactions on Geoscience and Remote Sensing* **1990**, *28*, 509–512. Conference Name: IEEE Transactions on Geoscience and Remote Sensing, doi:10.1109/TGRS.1990.572934.
39. Ojala, T.; Pietikäinen, M.; Mäenpää, T. Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns. *IEEE Trans. Pattern Anal. Mach. Intell.* **2002**, *24*, 971–987.
40. Petrovska, B.; Zdravevski, E.; Lameski, P.; Corizzo, R.; Štajduhar, I.; Lerga, J. Deep Learning for Feature Extraction in Remote Sensing: A Case-Study of Aerial Scene Classification. *Sensors* **2020**, *20*, 3906. Number: 14 Publisher: Multidisciplinary Digital Publishing Institute, doi:10.3390/s20143906.
41. Barbhuiya, A.A.; Karsh, R.K.; Jain, R. CNN based feature extraction and classification for sign language. *Multimedia Tools and Applications* **2021**, *80*, 3051–3069. doi:10.1007/s11042-020-09829-y.
42. Varshni, D.; Thakral, K.; Agarwal, L.; Nijhawan, R.; Mittal, A. Pneumonia Detection Using CNN based Feature Extraction. 2019 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT), 2019, pp. 1–7. doi:10.1109/ICECCT.2019.8869364.

43. Rippel, O.; Mertens, P.; König, E.; Merhof, D. Gaussian Anomaly Detection by Modeling the Distribution of Normal Data in Pretrained Deep Features. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 1–13. doi:10.1109/TIM.2021.3098381.
44. Yang, J.; Shi, Y.; Qi, Z. Learning deep feature correspondence for unsupervised anomaly detection and segmentation. *Pattern Recognit.* **2022**, *132*, 108874. doi:10.1016/J.PATCOG.2022.108874.
45. Reiss, T.; Cohen, N.; Bergman, L.; Hoshen, Y. PANDA: Adapting Pretrained Features for Anomaly Detection and Segmentation. IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021. Computer Vision Foundation / IEEE, 2021, pp. 2806–2814. doi:10.1109/CVPR46437.2021.00283.
46. Deng, J.; Dong, W.; Socher, R.; Li, L.; Li, K.; Fei-Fei, L. ImageNet: A large-scale hierarchical image database. 2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009), 20-25 June 2009, Miami, Florida, USA. IEEE Computer Society, 2009, pp. 248–255. doi:10.1109/CVPR.2009.5206848.
47. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet classification with deep convolutional neural networks. *Commun. ACM* **2017**, *60*, 84–90. doi:10.1145/3065386.
48. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *CoRR* **2018**, *abs/1804.02767*, [1804.02767].
49. Huang, G.; Liu, Z.; Maaten, L.V.D.; Weinberger, K.Q. Densely Connected Convolutional Networks. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR); IEEE Computer Society: Los Alamitos, CA, USA, 2017; pp. 2261–2269. doi:10.1109/CVPR.2017.243.
50. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. 2015, pp. 1–9. doi:10.1109/CVPR.2015.7298594.
51. Tan, M.; Le, Q. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. Proceedings of the 36th International Conference on Machine Learning; Chaudhuri, K.; Salakhutdinov, R., Eds. PMLR, 2019, Vol. 97, *Proceedings of Machine Learning Research*, pp. 6105–6114.
52. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the Inception Architecture for Computer Vision. 2016. doi:10.1109/CVPR.2016.308.
53. Szegedy, C.; Ioffe, S.; Vanhoucke, V.; Alemi, A.A. Inception-v4, inception-ResNet and the impact of residual connections on learning. Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence. AAAI Press, 2017, AAAI'17, p. 4278–4284.
54. Zoph, B.; Vasudevan, V.; Shlens, J.; Le, Q.V. Learning Transferable Architectures for Scalable Image Recognition. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); IEEE Computer Society: Los Alamitos, CA, USA, 2018; pp. 8697–8710. doi:10.1109/CVPR.2018.00907.
55. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770–778. doi:10.1109/CVPR.2016.90.
56. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv 1409.1556* **2014**.
57. Chollet, F. Xception: Deep Learning with Depthwise Separable Convolutions. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* **2016**, pp. 1800–1807.
58. Alam, S.; Sonbhadra, S.K.; Agarwal, S.; Nagabhushan, P. One-class support vector classifiers: A survey. *Knowl. Based Syst.* **2020**, *196*, 105754. doi:10.1016/J.KNOSYS.2020.105754.
59. Cheng, X.; Zhang, M.; Lin, S.; Zhou, K.; Zhao, S.; Wang, H. Two-Stream Isolation Forest Based on Deep Features for Hyperspectral Anomaly Detection. *IEEE Geosci. Remote. Sens. Lett.* **2023**, *20*, 1–5. doi:10.1109/LGRS.2023.3271899.

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