

Automated Analysis of Packaging Seal Failures Using Computer Vision Techniques

Análise Automatizada de Falhas na Selagem de Embalagens Utilizando Técnicas de Visão Computacional

Ana Osias^{1*}, Ian Otoni¹, Thiago L. Gomes¹, Michel M. Silva¹

Abstract: This work presents a computer vision-based method for the automated detection of sealing failures in carton packaging. In the juice industry, manual inspection of package flaps is destructive, time-consuming, and prone to human error, while defective sealing can lead to product leakage and economic losses. To address these challenges, we propose a two-stage approach: first, images of package flaps are preprocessed to accurately detect and align the flaps and detect the adhesive region; second, the adhesive region is segmented and classified into four categories: Aluminum, Background, Cardboard, and Cardboard-Aluminum using convolutional neural networks applied to image patches. Regions where aluminum appears in the same position on both flaps are identified as sealing failures. Experimental evaluation demonstrates that the method reliably detects defects, reducing reliance on manual inspection and providing an effective solution for automated quality control in industrial production.

Keywords: computer vision — convolutional neural networks — machine learning — automated defect detection

Resumo: Este trabalho apresenta um método baseado em visão computacional para a detecção automatizada de falhas de selagem em embalagens cartonadas. Na indústria de sucos, a inspeção manual das abas das embalagens é destrutiva, demorada e sujeita a erros humanos, enquanto falhas de selagem podem levar a vazamentos do produto e a prejuízos econômicos. Para enfrentar esse desafio, propomos uma abordagem em duas etapas: na primeira, as imagens das abas são pré-processadas para detectar e alinhar corretamente as superfícies e detectar a região adesiva; na segunda, a região adesiva é segmentada e classificada em quatro categorias: Alumínio, Fundo, Papelão e Papelão-Alumínio, por meio de redes neurais convolucionais aplicadas a trechos da imagem. Regiões em que o alumínio aparece na mesma posição em ambas as abas são identificadas como falhas de selagem. A avaliação experimental demonstra que o método detecta defeitos de forma confiável, reduzindo a dependência da inspeção manual e oferecendo uma solução eficaz para o controle de qualidade na produção industrial.

Palavras-Chave: visão computacional — redes neurais convolucionais — aprendizado de máquina — detecção automatizada de defeitos

¹ Departamento de Informática, Universidade Federal de Viçosa (UFV), Brazil

*Corresponding author: ana.osias@ufv.br

DOI: <http://dx.doi.org/10.22456/2175-2745.150833> • Received: 10/10/2025 • Accepted: 27/10/2025

CC BY-NC-ND 4.0 - This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License.

1. Introduction

Computer vision enables machines to perceive and interpret images and videos, serving as a cornerstone in the digitization of industrial systems. By leveraging machine learning and deep learning techniques, it allows the extraction of meaningful information from visual data, facilitating automation and decision-making. Its applications are wide-ranging, encompassing machine inspection, retail, 3D modeling, and real-time industrial quality control [1, 2, 3]. According to Vikas *et al.* [4], integrating computer vision into manufacturing is

a key step for Industry 4.0, improving efficiency, accuracy, and safety while enabling machines to interpret visual data for smarter and more connected production.

In this context, Brazilian companies in the juice sector seek to expand the adoption of computer vision technologies. Despite the high level of automation already in place, the packaging sealing stage remains a critical source of occasional failures. Such failures may arise from factors such as parameter adjustments in filling machines, cleaning of internal actuators, or defects in the packaging roll. When undetected,

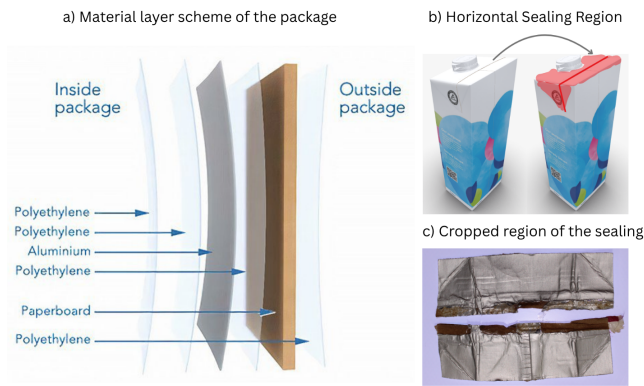


Figure 1. (a) Material layer scheme of the package. (b) Horizontal sealing region. (c) Cropped region of the sealing. Images (a) and (b) are adapted from Tetra Pak [5].

they compromise product quality, often resulting in juice leakage and economic losses. The current inspection procedure is limited and inefficient: the integrity of the packages is assessed through the sampling of a small number of units, using a destructive and manual process performed by an operator. At scheduled intervals, the operator arbitrarily selects some packages, discards their contents, and makes cuts in the section shown in Figure 1-b, separating the package flaps and obtaining a sample, as illustrated in Figure 1-c, to expose the adhesive region where the horizontal sealing can be observed.

The sealing defect occurs when, upon separating the flaps and exposing the sealing region, it is observed that the adhesive was unable to break the aluminum and expose the cardboard on at least one of the flaps, as shown in Figure 2. This happens because, as illustrated in Figure 1-a, the packages are composed of layers of aluminum, cardboard, and polymer; when the bonding is properly executed, the cardboard layer is exposed, whereas, otherwise, both flaps remain covered with visible aluminum.

The company reports difficulties in training employees to reliably identify sealing defects in packages. Furthermore, there is a strategic interest in implementing continuous data collection to support future analyses and to establish a structured record of performed tests. In this context, we propose

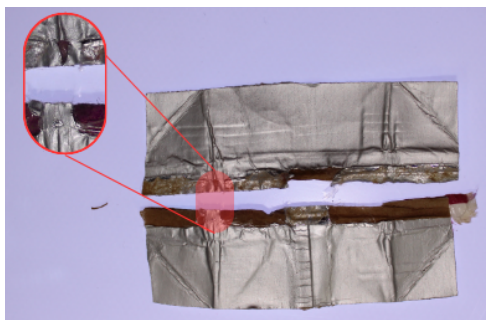


Figure 2. Example of adhesive failure. The highlighted region shows both **package flaps** with a small section of aluminum aligned in parallel, indicating insufficient bonding.

an automated computer vision-based technique that evaluates cropped and prepared images of package flaps to determine whether the sealing process was successfully executed. This approach addresses limitations of manual inspection, while enabling scalable, consistent, and data-driven quality control.

The scarcity of available failure samples poses a major challenge for training machine learning models to recognize defective packages. Although sealing defects generate significant costs, their low frequency of occurrence makes it impractical to construct, from scratch, a large dataset that would meet the requirements of data-hungry approaches such as deep learning. For this reason, we adopted an approach that combines digital image processing, which does not require large amounts of data, with the training of classifiers for patches of the materials present in the package, a more feasible alternative to collect and label within our context. This approach involves two main steps. The first consists of image preprocessing to correctly identify and position the two flaps of the package and to detect the adhesive application region. The second step involves defect region identification, in which a segmentation mask is created for the adhesive region, categorizing the classes: Aluminum, Background, Cardboard, and Both (Cardboard and Aluminum). The region is then analyzed to verify the occurrence of failures, that is, cases in which aluminum appears in the same position on both flaps.

The adoption of this automated technique, therefore, not only meets the immediate need for sealing evaluation but also lays the foundation for building a data repository that can support future artificial intelligence initiatives aimed at the continuous improvement of the production process.

2. Related Work

Automatic defect inspection in industrial processes has gained considerable attention, with various methods proposed to enhance detection efficiency and reliability.

The study by Diaz *et al.* [6] investigated the use of Convolutional Neural Networks (CNNs) for inspecting seal integrity in sterile packaging, a critical process for highly regulated products like medical devices. The authors released a public dataset and experimented with various architectures, including ResNet, DenseNet, and EfficientNet, which yielded promising results. A key finding was that segmentation models, despite their longer inference time, proved more robust for handling the low occurrence of defects and were less prone to overfitting compared to standard classification models.

In a broader context, Ren *et al.* [7] present a systematic review of the state of the art in defect detection based on machine vision. The paper discusses fundamental aspects such as optical illumination and image acquisition, as well as advanced image processing methods. The authors emphasize the growing impact of deep learning in classification, localization, and segmentation tasks, highlighting computer vision as a scalable and robust alternative to manual inspection, with strong potential to improve efficiency in production lines.

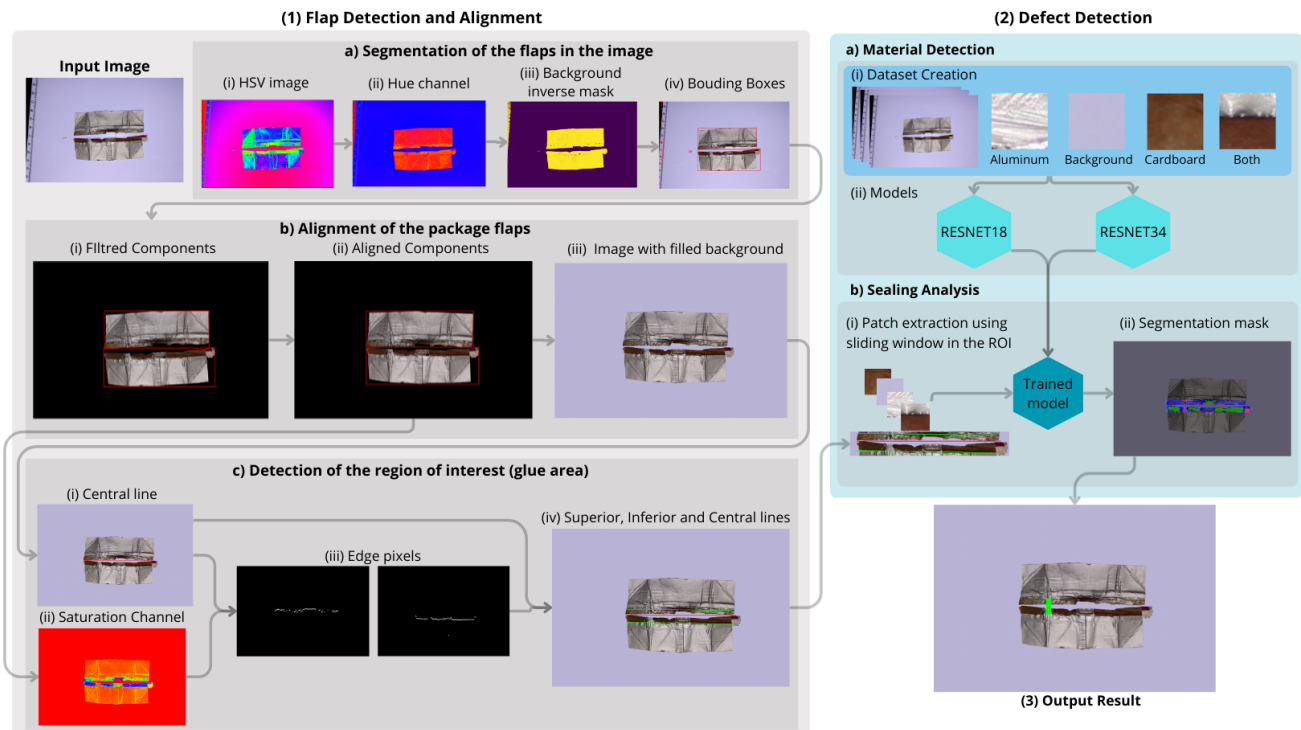


Figure 3. Overview of the proposed system comprising: (1) flap detection and alignment, (2) defect detection using material classification and sealing analysis, and (3) generation of the final aligned and segmented image.

From an applied perspective, Li *et al.* [8] propose a defect detection system for automotive high-pressure oil circuit seals, combining image preprocessing, feature point extraction using the SURF algorithm, and segmentation based on K-Means optimized by Particle Swarm Optimization. The method achieved an accuracy of 98% in detecting defects such as scratches and melting points, demonstrating high applicability in real industrial environments.

These studies emphasize the role of computer vision and machine learning in defect inspection. Similarly, our project focuses on low-occurrence defect detection with real-time, reliable performance.

3. Methods

We propose an automated computer vision-based method to assess the correctness of the sealing process in package flaps. The approach comprises two main stages, as illustrated in Figure 3. First, the flaps are detected, aligned and the adhesive application region is identified (Figure 3 (1)). Second, defect detection is performed through segmentation of the adhesive region and sealing analysis (Figure 3 (2)).

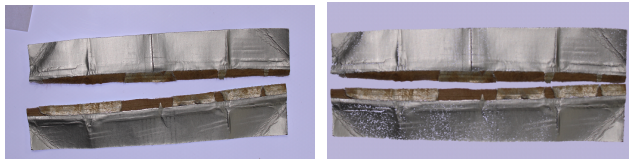
3.1 Flap Detection and Alignment

The captured images present variations in the orientation of the package flaps, depending on how the operator positioned them during this process, the capture process will be better described in the “Material Detection” subsection. To address this variability, an alignment step is necessary. This procedure

consists of identifying and adjusting the flaps so that they are positioned parallel to each other in the image. By standardizing the orientation, it becomes possible to consistently locate the glue region, which constitutes the main ROI for the algorithm. To this end, the preprocessing workflow, illustrated in Figure 3(1), is organized into three steps: (a) segmentation of the flaps in the image; (b) alignment of the package flaps; and (c) detection of the ROI (glue area).

Segmentation of the flaps in the image: The first step involves segmenting the regions corresponding to the package flaps. In the HSV color space, particularly in the Hue (H) channel, the packaging exhibits a clear distinction from the background. Accordingly, the algorithm converts the original image to HSV, as illustrated in Figure 3 (1a) subfigure (i), and isolates the Hue channel, as shown in Figure 3 (1a) subfigure (ii). A mask is then generated based on the Hue values, with values between 100 and 150 corresponding to the background. To obtain a mask of all regions that are not background, the inverse of this mask is created. The resulting mask is shown in Figure 3 (1a) subfigure (iii). The relatively uniform background facilitates the segmentation process. Finally, the contours and bounding boxes of the components are extracted using OpenCV’s *connectedComponentsWithStats* function, with the resulting bounding boxes illustrated in Figure 3 (1a) subfigure (iv).

Alignment of the package flaps: To accurately determine the position of the packaging flaps, we filtered the two com-



(a) Original image

(b) Image after alignment.

Figure 4. Left: Image of the flaps before processing. Right: Image of the flaps after processing, showing aligned flaps and a uniform background.

ponents with the largest area as illustrated in Figure 3 (1b) subfigure (i). With the information about the position of each part of the package, our approach consists of aligning these parts to identify the region where classification will be performed in the subsequent steps. To horizontally align each identified flap, the main orientation is estimated from the image moments, statistical measures that describe the shape and distribution of the component, which in this case are the package flaps. Based on this orientation, a rotation matrix is created to correct the inclination, which is then applied to both the component image and its mask. After rotation, all component references, such as the bounding box, are updated to reflect the new horizontal position. Finally, the flap located on the right was horizontally shifted towards the other flap, and the vertical adjustment, given by the difference between the lowest point of the lower flap and the highest point of the upper flap, was also calculated so that they could be aligned in this direction as well. The binary mask of the shifted component was updated to reflect its new position. Thus, the components became aligned, as shown in Figure 3(1b) subfigure (ii). During alignment, areas outside the components remain empty (black pixels). To address this, a real texture patch was replicated to fill the background, preserving the visual characteristics of the original image and avoiding missing information that could affect classification accuracy, as shown in Figure 3(1b) subfigure (iii). Figure 4 shows the image before and after alignment.

Detection of the region of interest (glue area): In the previous paragraph, we performed the alignment of the package flaps, which facilitates the detection of the region where the glue is applied. Up to this point, the flaps have been identified and aligned in the image; however, there was still no precise reference for the glue application area, the ROI. To detect this region, we started by tracing a line in the area between the upper and lower flaps. The central line between the two flaps was defined directly using the coordinates of the bounding box edges of the upper flap (lower edge) and the lower flap (upper edge), also taking into account their x -extremes. Since the bounding boxes are already aligned, this allows accurate detection of the central region. The obtained points were then used to fit a polynomial curve via the RANSAC algorithm [9], in order to represent the central line of the ROI, as shown in Figure 3(1c) subfigure (i). With the central line detected, we then identified the upper and lower lines that delimit the glue

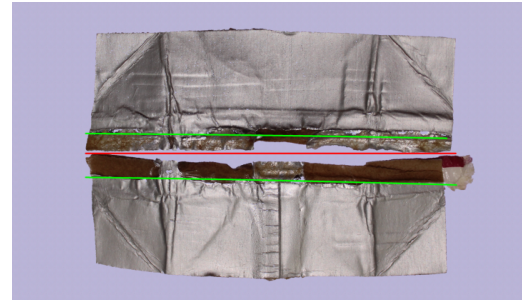


Figure 5. Segmented package flaps with the detected centerline (red) and the upper and lower boundaries (green) of the region of interest.

region. For this purpose, we used the Saturation (S) channel of the HSV color space, shown in Figure 3(1c) subfigure (ii), which highlights the glue application area, and applied the Canny edge detector [10]. From the central line, in each column, we selected the highest and lowest edge pixels in the edge mask, generating two masks: one with the upper pixels and another with the lower pixels, as shown in Figure 3(1c) subfigure (iii), which define the boundaries of the ROI. After extracting these edge pixels, each set was used to fit a polynomial curve using the RANSAC algorithm. The result is the smoothed upper and lower lines, represented by polynomial curves that delineate the glue region along the package, Figure 3(1c) subfigure (iii). The results for the upper, lower, and central lines can be observed in Figure 5.

3.2 Defect Detection

In manual inspection, proper adhesion causes the aluminum surface layer of the flaps to tear, exposing the underlying cardboard. Conversely, if adhesion fails in a section of both flaps, the aluminum remains visible in the corresponding areas, indicating insufficient bonding. Building on this observation, our defect detection method involves two steps: (a) detecting material within the packing and (b) analyzing the class distribution above and below the centerline (in the ROI).

3.2.1 Material detection

To perform segmentation with a classification model, a sliding window strategy was employed, generating crops across the entire ROI and classifying each pixel according to the learned material categories. To support this process, a dataset of patches representing the materials present in the packaging was created, and two classification models were trained to recognize these materials. The following paragraphs provide a detailed description of these procedures.

Dataset Creation Approximately seventy samples were provided by the partner company, consisting of randomly selected juice boxes taken directly from the production line. Each box was emptied, and either the upper or lower section was cut. The glued flaps were then carefully separated using a tool to expose the adhesive region. The flap pairs were then placed on an LED table to minimize shadow effects and photographed in

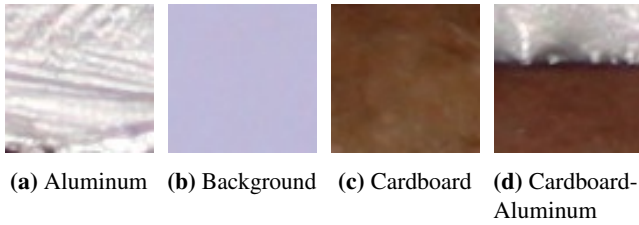


Figure 6. Patches of each class generated from the captured images.

the laboratory using a Canon DSLR EOS Rebel T7 camera in an indoor environment with the lights on. From these images, 64×64 -pixel patches were extracted and manually classified by the authors into four classes: **0 – Aluminum**, **1 – Background**, **2 – Cardboard**, and **3 – Cardboard-Aluminum**. Figure 6 shows examples of patches generated from the captured images. These patches were used to create datasets for training the classification models.

From these patches, a dataset was constructed with 423 patches per class, totaling 1,692 samples, Figure 3(2a) subfigure (i). To maintain a balanced dataset, the number of samples per class was limited based on the least frequent classes, which are Aluminum and Aluminum-Cardboard.

Models For performance comparison, two widely recognized convolutional neural network architectures were selected: ResNet18 and ResNet34, Figure 3(2a) subfigure (ii). The pre-trained weights for these models were obtained from the TorchVision library [11]. This version of the weights was trained on the ImageNet-1K dataset, which contains 1,000 object classes. These networks were chosen due to their proven effectiveness in image classification tasks, relatively low computational cost, and ease of implementation, features that are particularly advantageous given the available hardware limitations [12]. Selecting two distinct ResNet allows for evaluating the impact of network depth on model performance.

The data partitioning was made with the division: 70% for training, 15% for validation and 15% for test. The training was conducted for 30 epochs, using a learning rate of $1e-3$, batch size equal to 36 and the AdamW optimizer.

3.2.2 Sealing Analysis

Finally, with the image aligned and delimited by the upper and lower ROI lines, a sliding window-based segmentation method was applied to classify the different materials in the ROI pixel by pixel.

A fixed-size 64×64 window traversed the region bounded by the upper and lower lines, sliding horizontally and vertically with a defined stride. We conducted experiments using strides of 36 and 8 pixels. At each window position, a patch was extracted from the image, Figure 3(2b) subfigure (i), and processed by the trained neural network.

The predictions of each patch were accumulated into vote maps for each class. In this process, the classification assigned to the patch is added to a vote matrix for each pixel. Simultaneously, a second matrix records how many times each pixel

received a vote. This approach allows for the normalization of vote values across classes, enabling the assignment of the most voted class to each pixel. The final outcome is a mask generated by majority voting for the ROI, Figure 3(2b) subfigure (ii), where each pixel corresponds to the class to which it was most frequently assigned.

Finally, for the defect detection, a function was developed that, for each column of the ROI, analyzes the class distribution above and below the centerline. A column is flagged as a defect if it meets specific criteria indicative of a glue application failure: either aluminum is present in both the upper and lower parts, or aluminum is in one part and the "Cardboard-Aluminum" class is in the other. These detected defects are then highlighted with bounding boxes overlaid on the original image, Figure 3(3). This approach facilitates both the visualization and qualitative analysis of the results.

4. Results and Discussion

In this section, we present the experimental results obtained with the proposed approach. First, we analyze the performance of the classification models and their confusion matrices. Then, we illustrate the generation of segmentation masks under different stride configurations and evaluate the effectiveness of the defect detection process. Finally, we discuss some limitations and failure cases observed during the experiments.

4.1 Models

The mean metrics of the fine-tuned models are presented in Table 1. Both models exhibited consistent performance, with training accuracy above 99% and test accuracy, precision, recall, and F1-Score above 98%.

To assess whether there is a significant difference between the performances of the ResNet18 and ResNet34 architectures, we conducted nine training runs using different random seeds. For each seed, we recorded all main metrics of both models: accuracy, precision, recall, and F1-Score.

From these results, we constructed datasets consisting of the differences in each metric. The Shapiro–Wilk test [13] was then applied to verify whether these differences followed a normal distribution—a necessary condition for applying the t-test. Normality was confirmed. Subsequently, paired t-tests were applied to compare the mean differences in each metric. As discussed by Xu *et al.* [14], when two samples are related, as in cases where the same experimental units are evaluated by two distinct methods, the paired t-test is the most appropriate

Table 1. Mean metrics of the fine-tuned models.

Metric	ResNet-18	ResNet-34
Train Acc (%)	99.52	99.26
Test Acc (%)	98.82	98.03
Precision	0.9880	0.9810
Recall	0.9880	0.9800
F1-Score	0.9882	0.9804

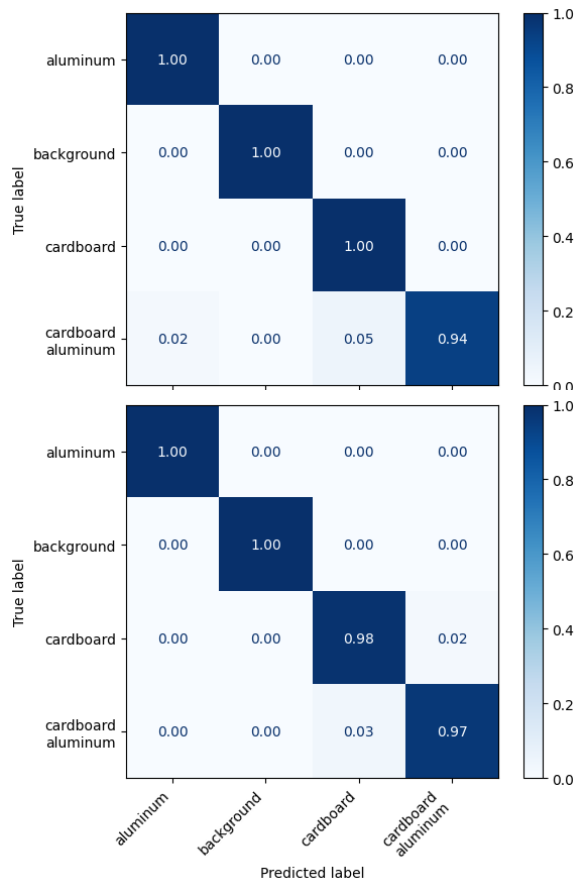


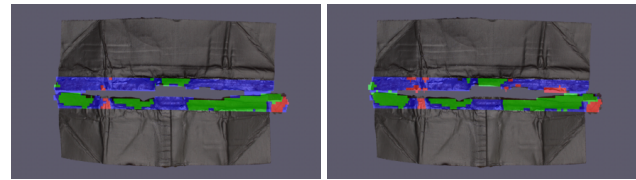
Figure 7. Confusion matrices for the ResNet-18 (top) and ResNet-34 (bottom) models. Matrices share the x-axis labels.

statistical procedure.

At a significance level of 5%, the tests indicated that there are no statistically significant differences between the metrics obtained by the two models.

Analysis of the confusion matrices for one of the seeds in Figure 7 indicates that the primary difficulty lies in distinguishing between the cardboard-aluminum class and the cardboard class. This confusion is expected, as the cardboard-aluminum class shares visual features with cardboard, which constitutes part of its structure. In the context of our problem, the most critical error occurs when a cardboard-aluminum sample is misclassified as cardboard, potentially resulting in an undetected packaging defect.

In the ResNet-18 model Figure 7-top, this type of misclassifications affected 7% of the cardboard-aluminum samples, with 5% classified as cardboard and 2% as aluminum. For ResNet-34 (Figure 7)-bottom, 3% of cardboard-aluminum samples were misclassified as cardboard. Additionally, ResNet-34 produced the opposite error, classifying 2% of cardboard samples as cardboard-aluminum. These confusion matrices were computed on the test partition of the dataset, corresponding to approximately three misclassified samples out of 254.



(a) ResNet-18 model

(b) ResNet-34 model

Figure 8. Pixel-wise classification masks generated using the ResNet-18 and ResNet-34 models with a stride of 32 pixels. Colors indicate the classes: red for aluminum, green for cardboard, and blue for cardboard-aluminum.

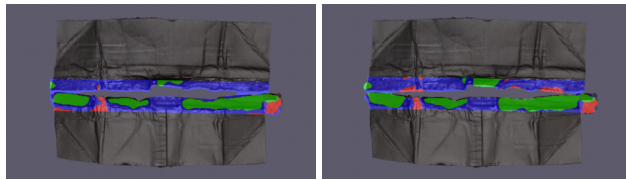
4.2 Segmentation Mask

To generate the masks, a sliding window approach was employed across the previously defined ROI, with a window size of 64×64 pixels for patch classification. For each pixel belonging to a patch, the predicted class results were accumulated through a voting system. We conducted experiments using two different strides for this sliding window. In one of them, the windows advanced with a stride of half a patch, i.e., 36 pixels, in both horizontal and vertical directions. In the second experiment, the window slid with a stride of one-eighth of the window size, i.e., 8 pixels per step in both directions. The effect of this variation is that, in the first case, inference is faster; however, pixels receive fewer votes for the class to which they belong.

4.2.1 Stride of 32 pixels

Using a stride of 32 pixels, we generated masks for each model. Figure 8a presents the mask generated for one of the samples using the ResNet-18 model, while Figure 8b shows the result obtained with the ResNet-34 model. The pixel-wise classification masks follow the following color scheme: red for aluminum, green for cardboard, and blue for cardboard-aluminum.

It should be noted that the visual results were similar, differing mainly in the detection of aluminum. It is important to emphasize that, for our problem, the correct classification of this material is particularly relevant. Considering the possible impacts of classification helps to illustrate this point. When a segment is erroneously classified as aluminum, a defect may be indicated in a package that is, in fact, adequate. This scenario can lead to unnecessary time spent on inspection and may even cause interruptions in the production process. Conversely, when a segment that should be identified as aluminum is classified as another material, there is a risk of validating an entire defective batch, in addition to masking potential failures in the company's machinery. For this reason, in the example discussed, the result obtained by ResNet-34 proves to be more appropriate than that of ResNet-18, even though both achieved satisfactory performance. Furthermore, it is observed that, in both cases, the defect was correctly detected despite this difference.



(a) ResNet-18 model

(b) ResNet-34 model

Figure 9. Pixel-wise classification masks generated using the ResNet-18 and ResNet-34 models with a stride of 8 pixels. Colors indicate the classes: red for aluminum, green for cardboard, and blue for cardboard-aluminum.

4.2.2 Stride of 8 pixels

Using a stride of 8 pixels, we also generated masks for each model. Figure 9a presents the mask generated for one of the samples using the ResNet-18 model, while Figure 9b shows the result obtained with the ResNet-34 model.

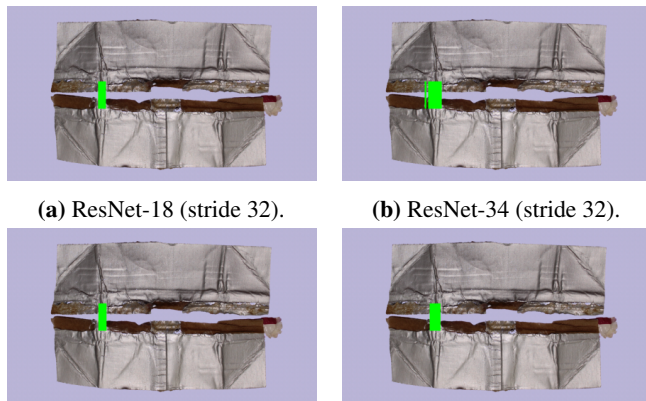
Although the process is more computationally expensive, the resulting masks were not significantly different from those obtained using the larger stride presented previously. However, it can be observed that, for the ResNet-18 model, more pixels were classified as belonging to the aluminum and cardboard-aluminum classes. For the ResNet-34 model, the differences are even more subtle.

4.3 Defect identification

The figure presenting the defect detection result was generated from the original image shown in Figure 2. For both models and both stride values, the algorithm was able to correctly identify the region where the defect occurs. The results obtained with a stride of 32 pixels are shown in Figures 10a and 10b, and with a stride of 8 pixels in Figures 10c and 10d.

In both cases, the result based on the mask generated by the ResNet-34 model considers a slightly larger region; nevertheless, the defect was correctly detected in all instances.

Cases without defects were also successfully processed by



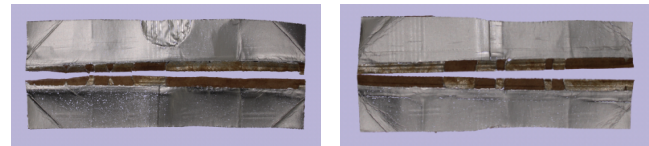
(a) ResNet-18 (stride 32).

(b) ResNet-34 (stride 32).

(c) ResNet-18 (stride 8).

(d) ResNet-34 (stride 8).

Figure 10. Comparison of defect detection results using ResNet-18 and ResNet-34 with stride 32 (top) and stride 8 (bottom). Detected defects are highlighted in green.



(a) Sample 1

(b) Sample 2

Figure 11. Defect detection results for two defect-free samples using ResNet-18 and ResNet-34.

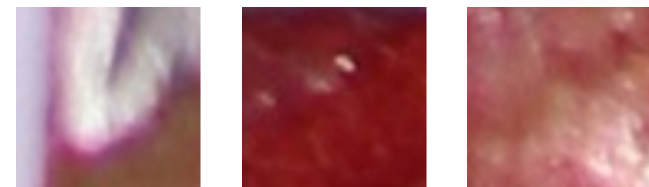
the algorithm, as shown in Figure 11. The same images are presented for both models and stride sizes, as the visual results were essentially identical. It is important to note that, in cases where a defect occurs, the algorithm marks the affected region with a rectangle indicating the detected defect. Additionally, the algorithm returns False when no defect is detected, as was the case for the samples shown in this figure.

4.4 Limitations

Despite the promising results, several aspects of our algorithm can still be improved. Concerning the classification models, the samples shown in Figures 12 and 13 correspond to misclassifications. It is important to note that these samples represent particularly challenging cases, where visual features of different classes are highly similar. For instance, distinguishing between the cardboard-aluminum and cardboard classes remains difficult due to overlapping texture and color characteristics. Additionally, variations in lighting, reflections on the aluminum surface, or slight deformations of the flaps may negatively affect model performance. Finally, the current dataset size and diversity may constrain the generalization capability of the models.

Concerning the alignment pipeline, some limitations can also be observed. The alignment is not perfect in certain samples, particularly with respect to the horizontal adjustment, as illustrated in Figure 11b. Currently, this alignment is performed using the centroid of the connected component. However, the method does not always yield precise results. Nevertheless, for all tested samples, this imperfection did not interfere with the final outcome.

Similarly, line detection is not always accurate across all cases. This limitation is more difficult to evaluate with the number of samples currently available, as it is not yet



(a) Cardboard-aluminum misclassified as aluminum.

(b) Cardboard-aluminum misclassified as cardboard.

(c) Cardboard-aluminum misclassified as cardboard.

Figure 12. Error examples produced by the ResNet-18.



(a) Cardboard misclassified as Cardboard-aluminum. (b) Cardboard-aluminum misclassified as cardboard. (c) Cardboard-aluminum misclassified as cardboard.

Figure 13. Error examples produced by the ResNet-34

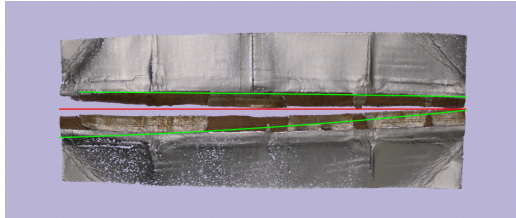


Figure 14. Example of inaccurate line detection. The green lines do not correctly delimit the glue region.

clear whether it will lead to significant issues. At present, these lines are estimated from the saturation mask, which reflects the region of the image covered by cardboard. More challenging samples, such as those with stronger aluminum traces, may not perform well under this approach. Figure 14 shows a region with increased aluminum–cardboard mixing produces a deviation in the detected line.

5. Conclusion

This work presented a computer vision-based solution for the automated detection of sealing failures in carton packaging. The approach integrates preprocessing, alignment, and CNN classification, achieving an accuracy above 98%. Experimental evaluation demonstrated that the method was able to perform well in cases different levels of packaging defects.

As future work, we plan to expand the dataset for validation in more varied scenarios and to implement a logging system for process traceability and continuous monitoring.

Acknowledgments

The authors would like to thank CAPES, CNPq, FAPEMIG and Tropical Industria de Alimentos S/A (Tial) for supporting this work. We are also grateful to Instituto de Inteligência Artificial e Computacional (Idata-UFV), FINEP (IA-AD-UFV #0284/22), and Cluster/UFV for providing GPU resources.

Author Contributions

Ana Osias was responsible for the entire project development, experiments, analyses, and manuscript writing; Thiago L. Gomes served as advisor and contributed through continuous supervision, methodological guidance, technical discussions,

and manuscript revision; Michel M. Silva served as advisor and contributed through continuous supervision, methodological guidance, technical discussions, and manuscript revision; Ian Otoni assisted in dataset construction and contributed to the manuscript writing.

References

- [1] ETTALIBI, A.; ELOUADI, A.; MANSOUR, A. Ai and computer vision-based real-time quality control: A review of industrial applications. *Procedia Computer Science*, v. 231, p. 212–220, 2024. ISSN 1877-0509. Disponível em: <https://www.sciencedirect.com/science/article/pii/S187705092302207X>.
- [2] ZHOU, L.; ZHANG, L.; KONZ, N. Computer vision techniques in manufacturing. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, v. 53, n. 1, p. 105–117, 2023. Disponível em: <https://doi.org/10.1109/TSMC.2022.3166397>.
- [3] SZELISKI, R. *Computer vision: algorithms and applications*. [S.l.]: Springer Nature, 2022.
- [4] VIKAS et al. Integration of computer vision systems in robotics and industry 4.0. In: *2024 International Conference on IoT, Communication and Automation Technology (ICICAT)*. [s.n.], 2024. p. 1492–1497. Disponível em: <https://doi.org/10.1109/ICICAT62666.2024.10923326>.
- [5] Tetra Pak. *Mantendo os alimentos seguros e nutritivos*. 2025. Acessado em: 17 set. 2025. Disponível em: <https://www.tetrapak.com/pt-br/insights/cases-articles/keeping-food-safe-and-nutritious>.
- [6] DIAZ, J. Z.; FAROOQ, M. A.; CORCORAN, P. Automatic inspection of seal integrity in sterile barrier packaging: A deep learning approach. *IEEE Access*, v. 12, p. 22904–22927, 2024. Disponível em: <https://doi.org/10.1109/ACCESS.2023.3348779>.
- [7] REN, Z. et al. State of the art in defect detection based on machine vision. *International Journal of Precision Engineering and Manufacturing-Green Technology*, Springer, v. 9, n. 2, p. 661–691, 2022. Disponível em: <https://doi.org/10.1007/s40684-021-00343-6>.
- [8] LI, X. et al. Surface defect detection of seals based on k-means clustering algorithm and particle swarm optimization. *Scientific Programming*, v. 2021, n. 1, p. 3965247, 2021. Disponível em: <https://doi.org/10.1155/2021/3965247>.
- [9] FISCHLER, M. A.; BOLLES, R. C. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, ACM New York, NY, USA, v. 24, n. 6, p. 381–395, 1981. Disponível em: <https://doi.org/10.1145/358669.358692>.
- [10] CANNY, J. A computational approach to edge detection. *IEEE Transactions on Pattern*

Analysis and Machine Intelligence, IEEE, PAMI-8, n. 6, p. 679–698, 1986. Disponível em: <https://doi.org/10.1109/TPAMI.1986.4767851>.

[11] PyTorch Foundation. *ResNet Model – Torchvision 0.20.0 documentation*. 2025. <https://pytorch.org/vision/main/models/resnet.html>. Accessed: 2025-10-08.

[12] HE, K. et al. Deep residual learning for image recognition. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. [s.n.], 2016. p. 770–778. Disponível em: <https://doi.org/10.1109/CVPR.2016.90>.

[13] SHAPIRO, S. S.; WILK, M. B. An analysis of variance test for normality (complete samples). *Biometrika*, [Oxford University Press, Biometrika Trust], v. 52, n. 3/4, p. 591–611, 1965. ISSN 00063444, 14643510. Disponível em: <https://doi.org/10.1093/biomet/52.3-4.591>.

[14] XU, M. et al. The differences and similarities between two-sample t-test and paired t-test. *Shanghai Archives of Psychiatry*, v. 29, n. 3, p. 184–188, 2017.