Article

Not peer-reviewed version

A CLIP-Based Framework to Enhance Order Accuracy in Food Packaging

Mattia Gatti *, Anwar Ur Rehman, Ignazio Gallo

Posted Date: 18 March 2025

doi: 10.20944/preprints202503.1328.v1

Keywords: Zero-shot learning; image-text matching; food classification; food recognition; industrial food packaging

 doi:10.20944/preprints202503.1328.v1

Article

A CLIP-Based Framework to Enhance Order Accuracy in Food Packaging

Mattia Gatti *, Anwar Ur Rehman and Ignazio Gallo

Department of Theoretical and Applied Science, University of Insubria, 21100 Varese, Italy
* Correspondence: mgatti3@uninsubria.it

Abstract: This study addresses the challenge of ensuring order accuracy in the dynamic environment of industrial food packaging through a novel zero-shot learning framework. The fundamental limitations of conventional systems, which rely heavily on pre-defined food categories, require a flexible approach capable of adapting to unseen and new food items. Our approach leverages the CLIP model for its efficient capability to semantically match text descriptions with image content, alongside YOLO's robust object detection abilities, to ensure accurate order fulfilment without prior knowledge of the food items. To assess the effectiveness of this approach, we introduced the Food Recognition dataset, comprising multi-compartment food packages with annotated food items, uniquely representing a variety of complex Italian recipes. Our CLIP-based approach can understand if a specific food name is represented by an image with a precision of 92.92% and a recall of 76.65% on the FR dataset, showcasing the model's effectiveness in recognizing and validating diverse food items in real-time scenarios. Furthermore, experiments conducted on 1000 entire food packages showed that our framework can detect whether a user's order matches the package contents with an accuracy of 85.86%. These results underline the potential of employing semantic image-text matching approaches to improve the efficiency of food packaging processes.

Keywords: zero-shot learning; image-text matching; food classification; food recognition; industrial food packaging

1. Introduction

Order accuracy refers to how closely shipments match customers' orders upon arrival. This includes having the right items in the order, the correct number of items, and no substitutions for ordered items [1]. Ensuring order accuracy is paramount for e-commerce enterprises as it profoundly influences customer satisfaction. Customers who receive the wrong item or quantity will likely be unhappy and may not shop with the business again. Enhancing order accuracy within an enterprise can be achieved through leveraging automated solutions capable of discerning whether some frames of a video of order processing align with the customer's request. Implementing continuous oversight of order processing operators enables swift error detection and rectification before shipping. However, this task is very challenging, especially in the industrial food packaging environment where the food offer changes fast. Moreover, ongoing monitoring must also address computational efficiency, as the execution time of automated solutions must not be a bottleneck that slows down the operator. Food detection is a crucial method in food computing, applying object detection approaches [2,3] in several practical situations, including automatic food package detection [4,5] and industrial food processing automation [6]. Traditional food detection approaches, trained on datasets with fixed categories, excel in recognizing predefined food classes. However, their utility is compromised in real-world scenarios like smart restaurants [7], where new food classes often appear, and latency is crucial to offer a proper service. Since these mapping-based approaches are supervised, they are easily biased towards seen classes, struggling to identify novel food categories. This limitation significantly restricts their application in dynamic real-industrial scenarios, where the diversity of food items

 doi:10.20944/preprints202503.1328.v1

continuously expands. One of the most known methods to deal with continuously expanding classes is the Contrastive Language–Image Pre-training (CLIP) [8] model. It introduced a different approach by combining image and text analysis in a zero-shot learning fashion, enabling it to understand and categorize unseen items effectively. Unlike traditional models that depend on fixed classes, CLIP's ability to interpret new classes through textual prompts allows for greater flexibility and adaptability in diverse environments. This makes CLIP an ideal solution for the ever-changing landscape of food varieties in food computing processes and automatic food detection industrial scenarios, offering a promising alternative to overcome the constraints of conventional food detection systems. The first part of this study presents the Food Recognition (FR) dataset, extending the work present in [4]. To create this dataset, we collected 15000 industrial images, detected food packages, and removed unnecessary images, resulting in 2510 high-resolution package images. Packages could have more than one compartment, each with its food content. All the contents must be detected to verify if a package matches an order. For this reason, we used the X-AnyLabeling tool [9] to efficiently annotate the food instances in the 2510 package images and generated a dataset for object detection and segmentation. YOLOv8 [10] and Mask-RCNN [11] were used on these images for food instance segmentation. The final FR dataset comprises 3850 cropped food images with a precise bounding box, segment information and name of the food. This dataset is very complex as it contains 280 food item classes, which are very traditional in Italian culture. Planeat, a food packaging company in Italy, provided all the images. This business was a good source for images since workers package food items according to individual client requirements, which vary significantly. Their workflow is defined in such a way that at the end of each order preparation, a photo of the operator's workbench with the packaging is taken by a camera placed in a nadir view (sensor looks straight down).

In the second part of this study, we propose a framework based on the CLIP model to recognize if packages match client orders, detecting whether there are no unexpected food items in the package. The most intuitive method would have been to compare the food in a given frame with all possible labels (food names), thereby determining if the best-matching label aligns with the order information. However, this approach presents its own challenges, including increased inference time and task difficulty with the expansion of classes. To address this, we propose a simplified task that involves judging whether a specific text prompt, i.e., the name of the expected food, is represented by an image. This approach was first evaluated using the popular Food-101 dataset, and after getting promising results from the benchmark dataset, it was evaluated on the FR dataset. The results of this study offer significant insights into the potential of our proposed solution in improving the efficiency and accuracy of order correctness evaluation in real-time food packaging. The summary of the contributions is as follows:

- Introduced a novel CLIP-based framework that employs zero-shot learning to accurately recognize and validate food items in food packaging, improving order accuracy.
- Achieved high precision (92.92%) and recall 76.65% in identifying food items from images, showcasing the model's robust image-to-text matching capabilities.
- Demonstrated an overall order accuracy of 85.86% in ensuring food packages match customer orders, highlighting the framework's potential.

2. Related Work

Traditional Computer Vision (CV) approaches for food recognition tried to solve the problem in a closed-set supervised fashion [12–14]. However, in recent years, Zero-shot learning (ZSL) has gained significant attention in the CV community due to the possibility of recognizing unseen classes [15–17]. OpenAI CLIP [8] is a popular model for various zero-shot problems in the literature. Regarding Zero-shot food detection, Chen et al. [18] proposed a framework to solve the problem of ingredient recognition in recipes. It leverages a multi-label convolutional neural network for known ingredient recognition and a multi-relational graph convolutional neural network (mRGCN) for unseen ingredients, combined with a multi-relational graph of different types of relations among ingredients, such as

 doi:10.20944/preprints202503.1328.v1

attribute, ingredient hierarchy, and co-occurrence in recipes. Li et al. [19] introduced ESE-GAN, a novel zero-shot food image classification that uses a generative adversarial network (GAN) to synthesize visual features of unseen classes. Another Zero-shot food detection approach was proposed by Zhou et al. [20] with the Semantic Separable Diffusion Synthesizer (SeeDS). The paper [21] introduces Zero-shot Food Detection (ZSFDet), a framework for detecting unseen food categories using multi-source graphs and a Region Feature Diffusion Model (RFDM) for robust feature synthesis. The authors present the FOWA dataset with 20,000+ images and 228 categories. Experiments demonstrate ZSFDet's superior performance, emphasizing the value of domain-specific knowledge for zero-shot detection. Another study [22] assesses ChatGPT-3.5 and ChatGPT-4 for extracting and linking food-related entities from text without training. ChatGPT evaluates in a Zero-shot manner and performs well in named-entity recognition (NER), particularly for recipes, but faces challenges in named-entity linking (NEL) due to difficulties in associating entities with semantic ontologies. The study highlights ChatGPT's potential for food information extraction but emphasizes the need for fine-tuning for reliable linking tasks. One more work [23] presents a framework combining graph convolutional networks (GCNs), a feature dictionary, and zero-shot learning (ZSL) to address fruit classification and segmentation with limited labeled data. The model identifies unseen fruit categories and integrates classification with segmentation. Experiments on multiple datasets, including the TLU-States dataset, show iCZSL outperforms existing methods, establishing a new benchmark for zero-shot learning in agricultural product recognition. Chen et al. [18] proposed a multi-relational graph convolutional network (mRGCN) for zero-shot ingredient recognition, achieving a recall of 84.3% for unseen ingredient classes but their approach lacks scalability to dynamic industrial workflows and does not address real-time performance requirements. Similarly, Li et al. [19] introduced ESE-GAN, a generative model for synthesizing features of unseen classes, achieving an accuracy of 82.4% in zero-shot food classification. However, their method focuses solely on classification tasks and does not integrate segmentation or end-to-end workflows necessary for applications like food packaging.

Furthermore, the order monitoring problem with food package recognition and food item segmentation is not discussed in the literature. The existing studies perform zero-shot classification without considering order information, making the task more difficult. For this reason, a simplified approach was defined for this scenario based on image-order information matching. The novelty of this work lies not in the simple integration of CLIP and YOLOv8 but in their purposeful application to address a critical industrial real-time automation challenge—food order validation in dynamic food packaging environments. Unlike existing zero-shot learning methods for food recognition, which primarily focus on static tasks such as ingredient detection or recipe recognition (e.g. multi-label CNNs or graph-based approaches), the proposed framework is uniquely designed for real-time industrial applications. Specifically, the integration of YOLOv8 for precise food item segmentation and detection with CLIP for zero-shot image-to-text matching is employed to ensure order accuracy in food packaging.

3. Material and Method

In the first step of the proposed framework, we leverage YOLO and Mask R-CNN for the segmentation of real-time industrial food package images. This allows for obtaining all the food items from a single frame. Second, the pipeline adopts a CLIP model to map the cropped food items to the order information in a zero-shot manner. This allows for finding the not-expected food items. A new Food Recognition (FR) dataset (Section 3.4) was made to train the segmentation models and to evaluate CLIP performance.

3.1. Food Segmentation

Step 1 of Figure 1 visually represents how the proposed framework's segmentation step works. The segmentation implicated processing images from the packaging process, often capturing a larger scene of the food packaging workspace. Crucial to this adaptation is the segmentation of food containers from the comprehensive scene to focus on the relevant portions of the image. Initially, the container is segmented with the procedure discussed in [4]. Then, from each container, the single food

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 18 March 2025 doi:10.20944/preprints202503.1328.v1

items are extracted. In our specific scenario, the extracted food items could be from one to three as there are 1-compartment, 2-compartment, and 3-compartment containers as shown by Figure 3.

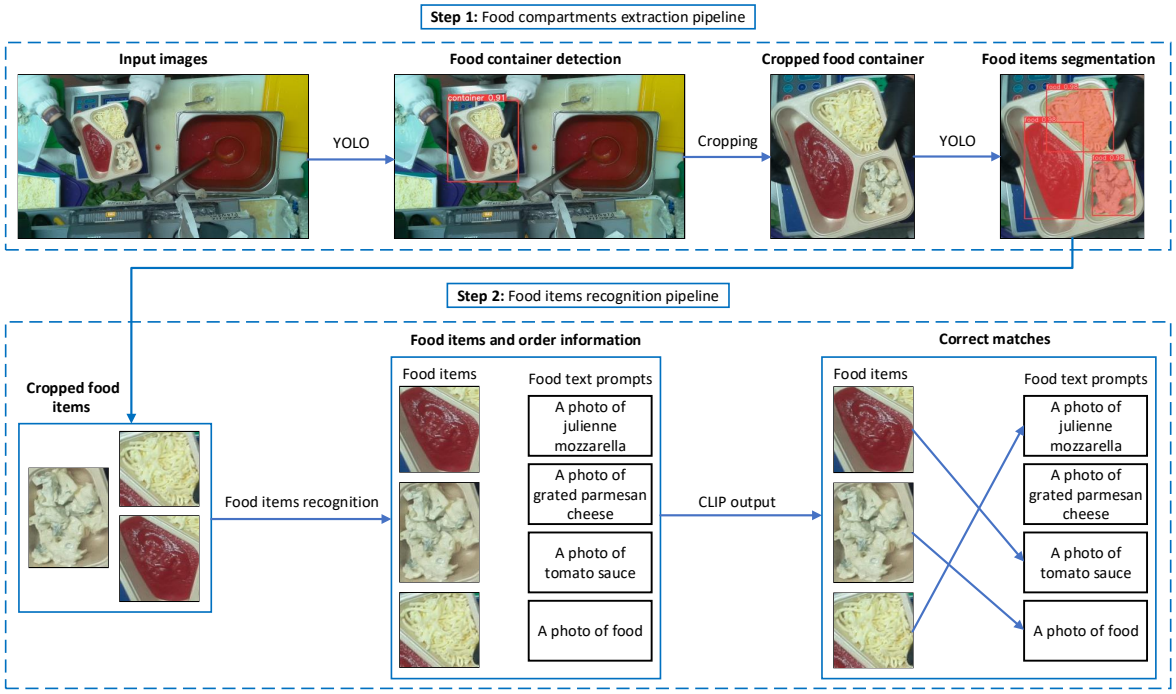

Figure 1. This study presents a two-stage process that utilizes a CLIP model to bridge the gap between visual food orders (images) and textual descriptions. First, a YOLO segmentation model separates the package from the larger scene image and then extracts individual food items. Second, a CLIP-based algorithm finds correct matches between food items and order information to find if the order is correct.

3.2. Contrastive Language-Image Pre-training (CLIP)

CLIP is designed to comprehend and align text prompts with visual content. Its basic purpose is to bridge the gap between language and visual representations, allowing the model to perform different visual activities, including classification, object detection, and more, without needing task-specific training data. This is especially valuable in zero-shot learning cases where the model must handle classes or tasks not specifically seen during training. **CLIP Architecture**: CLIP comprises two key components: an image and a text encoder. Both encoders focus on transforming their respective inputs into embeddings within the same high-dimensional space, easing direct comparison. **Image Encoder**: A convolutional neural network (CNN) or Vision Transformer (ViT) that maps an input image to a vector $\mathbf{v}$ in $\mathbb{R}^d$. **Text Encoder**: A Transformer-based model that converts a text prompt into a vector $\mathbf{t}$ in $\mathbb{R}^d$. **Contrastive Learning Objective**: The training goal of CLIP is to map the image and text embeddings so that relevant pairs are closer to each other than to non-relevant pairs. This is attained through a contrastive loss function:

$$L = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{\exp(\text{sim}(\mathbf{v}_i,\mathbf{t}_i)/\tau)}{\sum_{j=1}^{N}\exp(\text{sim}(\mathbf{v}_i,\mathbf{t}_j)/\tau)}$$

where: - N is the number of image-text pairs in a batch.

- $\text{sim}(\mathbf{v},\mathbf{t}) = \frac{\mathbf{v}^\top\mathbf{t}}{\|\mathbf{v}\|\|\mathbf{t}\|}$ denotes the cosine similarity between the image and text embeddings.
- τ is a parameter that scales the similarity scores.

Zero-Shot Learning with CLIP: In zero-shot learning, CLIP uses pre-trained encoders to classify images into unseen classes. Given an input image with suitable textual class descriptions, the model ensures the following steps:

1. Image encoder uses the input image to obtain its embedding $\mathbf{v}$.

 doi:10.20944/preprints202503.1328.v1

2. Text encoder encodes each class description to obtain a set of text embeddings $\{\mathbf{t}_1, \mathbf{t}_2, ..., \mathbf{t}_K\}$, where K is the number of potential classes.
3. Calculate each class's similarity score between the image and a text embedding.
4. Predict the class whose text embedding has the highest similarity score with the image embedding. The predicted class C^* is given by:

$$C^* = \arg\max_k \text{sim}(\mathbf{v}, \mathbf{t}_k)$$

The CLIP's ability to correlate textual descriptions with images enables it to generalize well across various domains. This procedure allows CLIP to accomplish classification tasks, including identifying food items, without having been explicitly trained in those classes.

3.3. CLIP for Food Recognition

In the industrial food packaging process, ensuring the accuracy of order correctness in real time needs a streamlined approach. Leveraging the CLIP model, we propose a baseline method that efficiently demonstrates whether the content of a food image aligns with the corresponding order information by comparing text descriptions and semantic embeddings of images. Given a dataset $\mathcal{D}$ of images x_i where $i \in \{1, 2, \ldots, N\}$, each image is linked with its correct label l_i describing the expected food content. The model's efficiency depends on comparing similarity scores between two types of textual descriptions and the image: the correct food label and a generic label, e.g. "food". The similarity score $s(x, l)$ provided by CLIP determines the alignment between an image x and a text prompt l.

The rule for validating the order correctness is mathematically expressed as:

$$c_i = \begin{cases} \text{True} & \text{if } s(x_i, l_i) > s(x_i, \text{"a photo of food"}) \\ \text{False} & \text{otherwise} \end{cases}$$

where c_i denotes the correctness of the order for image x_i.

This approach was initially validated using the Food-101 dataset, involving two primary tests to learn the model's predictive accuracy:

Correct Label Test: Each image x_i from the test split was evaluated to confirm if the CLIP predicts higher similarity for the correct label l_i over the generic label:

$$\text{Correct Prediction} = s(x_i, l_i) > s(x_i, \text{"a photo of food"})$$

An example of correct prediction from this is represented by Figure 2.

Incorrect Label Test: Each image x_i was tested with a purposely incorrect label l_i' to check if the CLIP appropriately scores the generic label higher:

$$\text{Correct Prediction} = s(x_i, \text{"a photo of food"}) > s(x_i, l_i')$$

Figure 2 represents an example of correct prediction from this test.

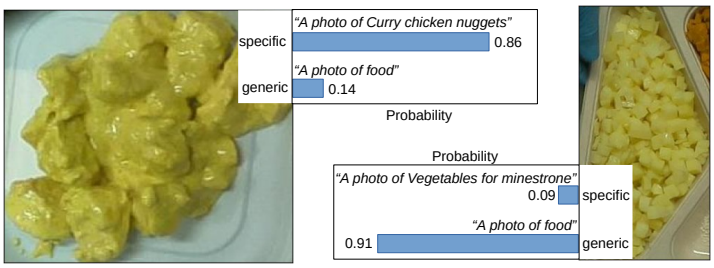

Figure 2. On the left, an example of a correct match from the correct label test; on the right, an example of a correct match from the incorrect label test.

 doi:10.20944/preprints202503.1328.v1

This simple approach can be used only with single-compartment packages, where a single food item has to be matched with a single food name. However, the output of the Food compartment extraction pipeline (Step 1 in Figure 1) could be more than one food item when there are multi-compartment packages. To extend the approach to N-compartment packages, N specific captions (the expected food items) must be used with the same generic text prompt discussed before. After CLIP inference, the food items with the generic text prompt as the best match are wrong and unexpected. This is represented by Step 2 in Figure 1. In this example, a person ordered Julienne mozzarella, grated parmesan cheese, and tomato sauce. However, nothing matched the parmesan cheese, meaning the order was wrong. Moreover, the food item matched with "A photo of food" is not expected.

Algorithm 1 represents the proposed algorithm to validate the orders.

Algorithm 1 Order validation algorithm

```
I: a set of food images
C: a set of expected food names (user order)
procedure ORDERVALIDATION(I, C)
    C ← C + "food"                          ▷ add the generic caption
    R ← true
    for each item i in I do
        scores ← CLIP(i, C)                 ▷ find the best descriptive caption for image i
        top ← argmax(scores)
        if C_top is not "food" then
            C ← C − C_top                   ▷ C_top can be removed from the food items to be found
        else
            R ← false                       ▷ Order is wrong
        end if
    end for
    return R
end procedure
```

Even if it is not specific in the algorithm, instead of using the label directly, the text prompt template is "a photo of {label}." CLIP paper states that this often improves performance over the baseline of using only the label text. For instance, just by using this strategy, they improved accuracy on ImageNet by 1.3% [8].

3.4. Dataset

The proposed framework was assessed on the existing Food101 [24] dataset and a newly created dataset that we called **Food Recognition (FR) Dataset**. The authors in [4] introduced the Food Packages (FP) dataset, a collection of 2000 images annotated with the detection boxes of various types of food packages shown in Figure 3. The FP dataset served as the foundation for this new dataset. From October 21, 2023, to November 30, 2023, we collected 15,000 images of the Planeat food packaging workspace and applied YOLOv7 (trained in [4]) to detect packages. This process yielded 10,000 images featuring different packages. After filtering out images with empty food packages and those where the package was not in an appropriate position, 2510 high-quality images of food packages remained. Then, the food inside the compartments of each package was annotated to create ground truths for object detection and segmentation tasks. We utilized the X-AnyLabeling tool [9] equipped with pre-trained weights from SOTA object detection and segmentation models, facilitating the semi-automatic labeling of the food instances. Some annotations were manually corrected because the automatic labeling provided by the X-AnyLabeling tool was not entirely accurate. During this annotation process, only one class named "food" was considered for all food instances inside the packages. Then, the labels of the FR dataset were exported in both YOLO and COCO formats. This provided a dataset for training a detection model to extract compartments' food content from a package. Then, the dataset was extended to the recognition task by cropping all the annotated foods and looking at the order information associated with each package. 3850 food images were extracted, each annotated with

 doi:10.20944/preprints202503.1328.v1

original Italian and translated English names. As the dataset is based on images from Planeat, which is based in Italy, it can cover 280 food products that are common in that country. This provided data for the correct and incorrect label tests discussed in the previous section. The source package's ID was stored for each cropped food item image. This made it possible to take all the food items belonging to the same package and perform the matching of these images with the order information, allowing the evaluation of the proposed Algorithm 1.

The proposed framework makes use of standard tools such as CLIP and YOLOv8, whose integration is specifically tailored to address the unique challenges in industrial food packaging environments and offers distinct advantages over previous methods. Traditional zero-shot learning approaches to food recognition, such as those using multi-label CNNs or graph-based networks, excel in static settings such as ingredient recognition or recipe classification, but lack scalability and efficiency for real-time industrial workflows. In contrast, our framework adapts these tools for a practical, end-to-end pipeline that spans food container detection, food item segmentation, and order verification.

This study recognizes the importance of analyzing failure cases to fully understand the limitations of the proposed methodology. During the evaluation, we found that the framework faced challenges with ambiguous food items, especially those with visual similarities, such as different types of cheese or sauces. In such cases, CLIP's semantic embedding space sometimes failed to accurately distinguish these items, resulting in false matches or lower confidence scores.

Figure 3. Images of different food containers with (a) one compartment, (b) two compartments, and (c) three compartments available in the FP dataset and used in our FR dataset.

4. Experiments

The first part of this study aims to evaluate the efficacy of two SOTA deep learning models: YOLOv8 medium [10] and Mask R-CNN [11]. The first experiments involved training YOLOv8 and Mask R-CNN models on 2260 annotated food images and testing on the remaining 250 of the new FR dataset. This was done to determine which model would better identify and segment objects of interest (food). The parameters for the segmentation experiments were carefully chosen through trial, to optimize the models' performance: a batch size of 16, training for 100 epochs, employing the SGD optimizer, and setting the Intersection Over Union (IoU) threshold at 0.7. To refine our approach further, we tested two distinct backbones for each model: ResNet [25] for Mask R-CNN and CSPDarknet53 [26] for YOLOv8. These two backbones were chosen because they tend to be a good compromise between accuracy and efficiency. After a comprehensive comparison, YOLOv8 was selected due to its superior performance on the test set in evaluation metrics, particularly Precision (P), Recall (R), and mean Average Precision (mAP), and its results are shown in Table 1

The second part of this study assesses the efficacy of the proposed CLIP-based order validation approach. Of all the available pre-trained CLIP models provided by OpenAI, the one with ViT-B/32 as an encoder was used. The proposed CLIP assessment consisted of two phases: evaluating the CLIP baseline approach, followed by the final algorithm proposal. The effectiveness of the baseline approach was evaluated utilizing the previously discussed correct label and incorrect label tests. We ran them on the test split of the Food101 dataset, which comprises 25250 images, and then on the 3850 cropped food items of the FR dataset with both Italian and English labels, and evaluation results are presented in Tab. Table 2. As the algorithm is built upon the baseline approach, it was necessary to obtain reliable metrics on the baseline approach before the algorithm definition. The successively built Algorithm 1 was evaluated using all the two and three compartments packages and the associated user orders from

 doi:10.20944/preprints202503.1328.v1

the FR dataset. For each container, all the food items were extracted, and the algorithm was evaluated based on its ability to find the correct match between the food names in the order information and the food item images. Results are shown in Table 3. An example of prediction made by our framework is represented by Figure 4. By using an Nvidia Quadro RTX 5000, the pipeline achieved an execution time of $\sim 60ms$ for the segmentation step (Step 1 in Figure 1) and then $\sim 100ms$ for the food recognition step (Step 2 in Figure 1), resulting in a total of $\sim 160ms$ for the whole pipeline.

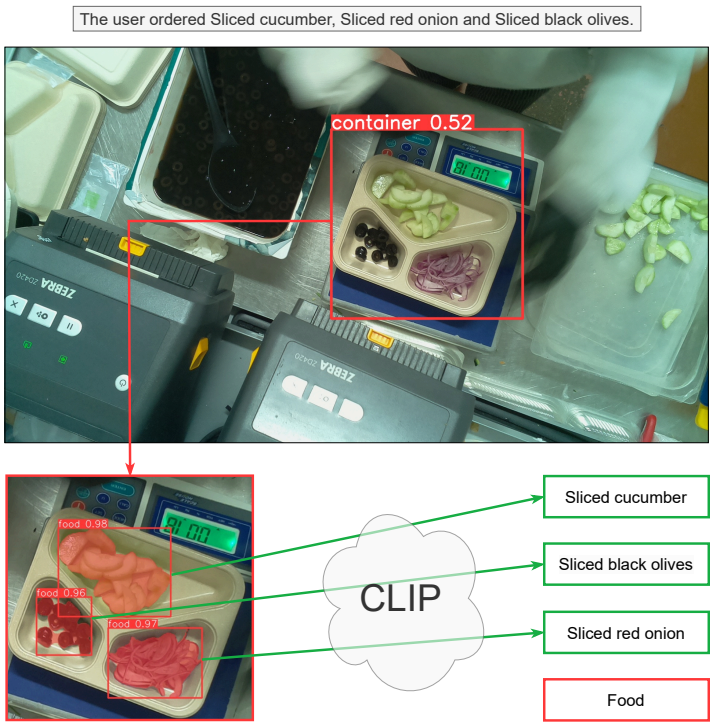

Figure 4. An example of a food order where the client requested sliced cucumber, sliced red onion, and sliced black olives. The top image is a real-time food packaging scenario given as input to the proposed framework. It detected all the requested items in the food package (bottom image) and identified the order as correct since nothing matched the caption "food".

5. Results and Analysis

YOLOv8 model demonstrated remarkable results in Tab. Table 1, achieving a segmentation mask Precision(P) and Recall(R) at 99.0% and 98.90%, respectively,% with a mAP0.50 of 98.90% and a mAP0.50:0.95 of 95.90% (see the first row of Table 1). These results underscore the model's exceptional ability to detect and segment objects within the dataset accurately. Conversely, the performance of Mask R-CNN, while commendable, did not match that of YOLOv8. It achieved 98.20% and 84.00% P and R with 84.40% mAP0.50 and 84.30% mAP0.50:0.95, respectively (see the second row of Table 1). This resulted in the decision to proceed with YOLOv8 as the segmentation model for the proposed framework.

Regarding the food recognition step, the baseline approach achieved excellent results in the Food101 benchmark dataset, achieving a P and R of 96.31% and 94.43%, respectively (first row in Table 2). For this test, the true negatives (TN) were the matches to be predicted as correct, while the true positives (TP) were those to be predicted as wrong. The achieved results were the first sign that this baseline approach could be viable even in a more complex dataset. This was further confirmed by the results in the FR dataset, achieving a P and R of 92.92% and 76.75 (second row in Table 2). A lower recall is probably due to the CLIP embedding space, which doesn't properly map certain complex Italian food names. This means that given a food image and the expected food name, the method can tell whether it is a good match. These results paved the way for the definition of a more complex algorithm to detect whether a user order (a set of food names) is described by a set of food images. The proposed order validation algorithm achieved good results, achieving an accuracy of 85.86% in

 doi:10.20944/preprints202503.1328.v1

matching cropped food images from packages with the user-ordered food names (Table 3). The results showed that this algorithm can accurately detect an operator's mistake during order processing in a food packaging business. In addition, the fast inference time of the proposed methodology allows for continuous monitoring and real-time correction. All metrics can be improved by creating a larger food dataset to fine-tune CLIP, but contrastive learning requires many images with many different labels, and building such a generic dataset is difficult.

The two recognition experiments involving the FR dataset were run using Italian and English food names. Both obtained better metrics when the labels were English translations rather than the original Italian food names. However, this behaviour was expected since CLIP is trained using English captions. Multilingual CLIP [27] can improve results when text descriptions are not in English, improving the utility of the proposed pipeline for international applications.

The proposed approach bridges the gap between theoretical advances in zero-shot learning and its practical implementation in an industrial setting. By targeting unseen classes, a key limitation of traditional supervised models, and optimizing the pipeline for real-time performance, we provide a scalable solution that meets the stringent accuracy and efficiency requirements of the food packaging industry. Importantly, our framework is the first to address the entire workflow from food container detection to food item segmentation to order verification, providing a holistic solution for industrial automation. The use of standard yet state-of-the-art methods such as CLIP and YOLOv8 is a deliberate choice, as it ensures the robustness and generalizability needed to tackle real-world industrial problems, effectively highlighting a key innovation over previous zero-shot food recognition techniques that lack such practical applications.

Table 1. Food detection results obtained with YOLOv8 and Mask R-CNN. It is clear that YOLOv8 outperforms the Mask R-CNN in all evaluation metrics.

Models	Images	Labels	P	R	mAP@0.5	mAP@0.5:0.95
YOLOv8	250	food	99.00%	98.90%	98.90%	95.90%
Mask R-CNN	250	food	98.20%	84.00%	84.40%	84.30%

Table 2. Results of the baseline approach on Food101 dataset and our dataset. Metrics were obtained using the mean between the correct and incorrect label tests. English captions showed better results than original Italian food names (as expected).

Dataset	Model	Images	Accuracy	Precision	Recall
Food101	CLIP	25250	95.40%	96.31%	94.43%
FR (EN)	CLIP	3850	85.40%	92.92%	76.65%
FR (IT)	CLIP	3850	71.40%	69.64%	75.87%

Table 3. Results of the proposed order validation algorithm. Accuracy refers to the number of correct matches between the compartments' content and the user order information. English captions showed (as expected) better results than original Italian food names.

Dataset	Model	Containers	Accuracy
FR (EN)	YOLO + CLIP	1000	85.86%
FR (IT)	YOLO + CLIP	1000	58.71%

6. Conclusion

This study presents a CLIP-based framework to monitor order accuracy in industrial food packaging using zero-shot learning for food item recognition and package content verification. This work

 doi:10.20944/preprints202503.1328.v1

also constructed a new Food Recognition dataset (FR) comprising 2510 multi-compartment food packages with 3850 food items images. The proposed framework achieved 92.92% and 85.86% precision and accuracy in the food matching with text prompts and for order validation using the FR dataset, respectively. In addition, the overall order accuracy for matching the food package contents with the clients' orders shows potential improvements in customer satisfaction and operational automation in e-commerce environments. The proposed problem of zero-shot order verification has never been discussed in the literature. However, the proposed methodology outperforms similar methods regarding food recognition and classification. On the other hand, experimental results indicate that the proposed CLIP-based framework could be a starting point for other zero-shot monitoring models, opening the way to assisted packaging operations even outside the food environment. In addition, this work will be improved in the future by increasing scalability to improve metrics on a broader range of food classes, optimizing computational efficiency for use in large-scale production lines, and improving robustness to constraints such as the use of Italian-language food names (captions). These improvements will strengthen the practical utility of the framework and promote its application in different industrial contexts.

References

1. Mentzer, J.T.; Flint, D.J.; Kent, J.L. Developing a logistics service quality scale. *Journal of Business logistics* **1999**, *20*.
2. Gallo, I.; Rehman, A.U.; Dehkordi, R.H.; Landro, N.; La Grassa, R.; Boschetti, M. Deep object detection of crop weeds: Performance of YOLOv7 on a real case dataset from UAV images. *Remote Sensing* **2023**, *15*, 539.
3. Rehman, A.U.; Gallo, I. Cross-pollination of knowledge for object detection in domain adaptation for industrial automation. *International Journal of Intelligent Robotics and Applications* **2024**, pp. 1–19.
4. Rehman, A.U.; Gallo, I.; Lorenzo, P. A Food Package Recognition Framework for Enhancing Efficiency Leveraging the Object Detection Model. In Proceedings of the 2023 28th International Conference on Automation and Computing (ICAC). IEEE, 2023, pp. 1–6.
5. Junyong, X.; Kangyu, W.; Hongdi, Z. Food packaging defect detection by improved network model of Faster R-CNN. *Food and Machinery* **2023**, *39*, 131–136.
6. Selvam, P.; Koilraj, J.A.S. A deep learning framework for grocery product detection and recognition. *Food Analytical Methods* **2022**, *15*, 3498–3522.
7. Aguilar, E.; Remeseiro, B.; Bolaños, M.; Radeva, P. Grab, pay, and eat: Semantic food detection for smart restaurants. *IEEE Transactions on Multimedia* **2018**, *20*, 3266–3275.
8. Radford, A.; Kim, J.W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. Learning transferable visual models from natural language supervision. In Proceedings of the International conference on machine learning. PMLR, 2021, pp. 8748–8763.
9. Wang, W. Advanced Auto Labeling Solution with Added Features. https://github.com/CVHub520/X-AnyLabeling, 2023.
10. Chaurasia, G.J.A.; Qiu, J. YOLOv8 Instance Segmentation. In Proceedings of the https://github.com/ultralytics/ultralytics, 2023, pp. 000–000.
11. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask r-cnn. In Proceedings of the Proceedings of the IEEE international conference on computer vision, 2017, pp. 2961–2969.
12. Kagaya, H.; Aizawa, K.; Ogawa, M. Food Detection and Recognition Using Convolutional Neural Network. In Proceedings of the Proceedings of the 22nd ACM International Conference on Multimedia, New York, NY, USA, 2014; MM '14, p. 1085–1088. https://doi.org/10.1145/2647868.2654970.
13. Subhi, M.A.; Md. Ali, S. A Deep Convolutional Neural Network for Food Detection and Recognition. In Proceedings of the 2018 IEEE-EMBS Conference on Biomedical Engineering and Sciences (IECBES), 2018, pp. 284–287. https://doi.org/10.1109/IECBES.2018.8626720.
14. Khan, S.; Ahmad, K.; Ahmad, T.; Ahmad, N. Food items detection and recognition via multiple deep models. *Journal of Electronic Imaging* **2019**, *28*, 013020. https://doi.org/10.1117/1.JEI.28.1.013020.
15. Bretti, C.; Mettes, P. Zero-shot action recognition from diverse object-scene compositions. *arXiv preprint arXiv:2110.13479* **2021**.
16. Thong, W.; Snoek, C.G. Bias-awareness for zero-shot learning the seen and unseen. *arXiv preprint arXiv:2008.11185* **2020**.

 doi:10.20944/preprints202503.1328.v1

17. Huang, H.; Chen, Y.; Tang, W.; Zheng, W.; Chen, Q.G.; Hu, Y.; Yu, P. Multi-label zero-shot classification by learning to transfer from external knowledge. *arXiv preprint arXiv:2007.15610* **2020**.
18. Chen, J.; Pan, L.; Wei, Z.; Wang, X.; Ngo, C.W.; Chua, T.S. Zero-Shot Ingredient Recognition by Multi-Relational Graph Convolutional Network. *Proceedings of the AAAI Conference on Artificial Intelligence* **2020**, *34*, 10542–10550. https://doi.org/10.1609/aaai.v34i07.6626.
19. Li, G.; Li, Y.; Liu, J.; Guo, W.; Tang, W.; Liu, Y. ESE-GAN: Zero-Shot Food Image Classification Based on Low Dimensional Embedding of Visual Features. *IEEE Transactions on Multimedia* **2024**, pp. 1–11. https://doi.org/10.1109/TMM.2024.3353457.
20. Zhou, P.; Min, W.; Zhang, Y.; Song, J.; Jin, Y.; Jiang, S. SeeDS: Semantic Separable Diffusion Synthesizer for Zero-shot Food Detection. In Proceedings of the Proceedings of the 31st ACM International Conference on Multimedia. ACM, Oct. 2023, MM '23. https://doi.org/10.1145/3581783.3612661.
21. Zhou, P.; Min, W.; Song, J.; Zhang, Y.; Jiang, S. Synthesizing Knowledge-enhanced Features for Real-world Zero-shot Food Detection. *IEEE Transactions on Image Processing* **2024**.
22. Ogrinc, M.; Koroušić Seljak, B.; Eftimov, T. Zero-shot evaluation of ChatGPT for food named-entity recognition and linking. *Frontiers in nutrition* **2024**, *11*, 1429259.
23. Tran-Anh, D.; Huu, Q.N.; Bui-Quoc, B.; Hoang, N.D.; Quoc, T.N. Integrative zero-shot learning for fruit recognition. *Multimedia Tools and Applications* **2024**, pp. 1–23.
24. Bossard, L.; Guillaumin, M.; Van Gool, L. Food-101 – Mining Discriminative Components with Random Forests. In Proceedings of the European Conference on Computer Vision, 2014.
25. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.
26. Wang, C.Y.; Liao, H.Y.M.; Yeh, I.H.; Wu, Y.H.; Chen, P.Y.; Hsieh, J.W. CSPNet: A New Backbone that can Enhance Learning Capability of CNN, 2019, [arXiv:cs.CV/1911.11929].
27. Carlsson, F.; Eisen, P.; Rekathati, F.; Sahlgren, M. Cross-lingual and Multilingual CLIP. In Proceedings of the Proceedings of the Thirteenth Language Resources and Evaluation Conference; Calzolari, N.; Béchet, F.; Blache, P.; Choukri, K.; Cieri, C.; Declerck, T.; Goggi, S.; Isahara, H.; Maegaard, B.; Mariani, J.; et al., Eds., Marseille, France, Jun. 2022; pp. 6848–6854.