Comparing machine learning vs. humans for dietary assessment

[1]Maryam Abbasi, [2]Cristina Wanzeller, [3]Filipe Cardoso, and [2]Pedro Martins

[1] University of Coimbra, Coimbra, Portugal
maryam@dei.uc.pt
[2] CISeD - Research Centre in Digital Services, Polytechnic of Viseu, Viseu, Portugal
pedromom@estgv.ipv.pt, cwanzeller@estgv.ipv.pt
[3] Polytechnic Institute of Coimbra, Coimbra, Portugal
filipe@isec.pt

Abstract. Due to the availability of large-scale datasets (e.g., ImageNet, UECFood) and the advancement of deep Convolutional Neural Networks (CNN), computer vision image recognition has evolved dramatically. Currently, there are three major methods for using CNN: starting from scratch, using a pre-trained network off the shelf, and performing unsupervised pre-training with supervised changes. When it comes to those with dietary restrictions, automatic food detection and assessment are critical.In this research, we show how to address detection difficulties by combining three CNNs. The different CNN architectures are then assessed. The amount of parameters in the examined CNN models ranges from 5,000 to 160 million, depending on the number of layers. Second, the various CNNs under consideration are assessed based on dataset sizes and physical image context. The results are assessed in terms of performance vs. training time vs. accuracy. Finally, the accuracy of CNNs is investigated and examined using human knowledge and classification from the human visual system (HVS). Finally, additional categorization techniques, such as bag-of-words, are considered to solve this problem.Based on the findings, it can be concluded that the HVS is more accurate when a data set comprises a wide range of variables. When the dataset is restricted to niche photos, the CNN outperforms the HVS.

Keywords: CNN, GoogLeNet, Inception, ResNet, dietary

1 Introduction

In the fields of image categorization [1] and object identification accuracy [3], Convolutional Neural Networks (CNNs) have advanced tremendously. When comparing object detection to image classification, object detection is a more difficult task that necessitates the use of more complex algorithms [13] due to the necessity for multistage pipelines that are slow and inefficient. Solutions make a trade-off between speed and precision.

CudaConvNet, Torch, Theano, and Caffe4 are just a few of the machine learning and CNN libraries available. MatConvNet, a MATLAB package that

implements CNN's oriented to computer vision, has been applied. The simple learning method used by CNN, which relies on convolutions and rectifications, is far from simple. The reason for this is that a functioning balance-trained network requires learning by modifying coefficients (backpropagation) from huge amounts of data (i.e., millions of images). MatConvNet, like other tools, is optimized for this task by applying a set of optimizations that allow automatic parallelism and support the use of GPUs to speed up certain mathematical computations (using CUDA DevKit and a compatible NVIDIA GPU).

Mobile devices with good computational qualities are important in computer vision and can help with better and healthier lifestyle dietary assessments. The initial step in food recognition is to automatically identify the dish containing the meal while ignoring everything else. Then, using the semantics (e.g., rice with beans, egg yolk, and white), separate the food parts in the dish and categorise the food. Eating habits can be improved with any application that tracks the calories taken in food, advising healthier lifestyle selections based on the retrieved data. Manual food identification can be replaced by automatic classification by just aiming a smartphone's camera at the food plate, thanks to the combination of CNNs with current mobile computer capacity. As a result, it's crucial to investigate whether CNNs can beat people and whether they're more efficient than other traditional methods to the problem.

We examine three Convolutional Neural Networks Applied to Food Detection and Classification, GoogLeNet [11], Inception-v3 [12], and ResNet-101 [5], using real-world food picture data sets (UEC-256 and Food-101). We studied and assessed alternative CNN architectures using the food dataset UECFood-256, using the same setup parameters when training the networks from scratch. The performance of all tested CNNs is discussed and rated based on the training that has been completed.

By treating image attributes as terms, the bag-of-words paradigm can be applied to the representation of images in computer vision. A bag of words is a sparse vector of term frequency counts in text classification; in other words, a sparse histogram of the vocabulary [6]. The same configuration is evaluated with CNNs, specifically GoogLeNet, Inception-v3, and Resnet101, as a more state-of-the-art deep-learning technique.

Furthermore, the accuracy of CNN's computer vision approach is compared to human vision-based classification, based on a study group of around 100 university students. It also entails learning new food dishes, such as European cuisine (Asian cuisine is unknown), and then identifying them. The designs of the bag-of-words pipeline and CNN's are described in this work, as well as the survey that was designed for training and assessing the food identification capacity of a selected set of individuals.

All classes, six epoch training, demonstrate an accuracy of 70.68 percent compared. 80.6 percent HVS based on a comprehensive food dataset. When the CNN is trained for only 16 food classes (the same as in the HVS trial), it achieves an accuracy of 89.89 percent. CNN's accuracy improves to 93.86 percent when the number of epochs is increased from 6 to 20.

This paper is organized as follows. Section 2, makes a short resume of the related work in the field. Section 3 describes the global experimental setup used to test the CNN. Section 4 shows obtained results and concludes. Section 5, concludes the work.

2 Related work

Object recognition entails extracting features from photos that are represented using a certain model. The features are then passed into a classifier, which recognizes the picture objects, which in this case is food. There are two types of features: global (e.g., color histograms) and local (e.g., circular shapes) (e.g., pixel color, SIFT features). The choice of such elements has a significant impact on the final accuracy. Many combinations are possible and have been proposed. Color and texture are blended, and visual elements such as contour, motion, texture, and color are integrated [9].

Three ResNet designs are researched and compared in [2]. First, Inception-ResNet-v1 is a hybrid version of Inception with a computational cost similar to Inception-v3. Second, there's inception-ResNet-v2, a more expensive hybrid Inception version with better recognition performance. Last but not least, Inception-v4, a pure Inception variation with no residual connections that performs similarly to Inception-ResNet-v2. The authors investigate how the addition of residual connections improves the Inception architecture's training time considerably.

The authors suggest an autonomous food picture identification system for recording people's eating patterns in [10]. In their experiments, they used the Multiple Kernel Learning approach to adaptively combine multiple image elements such as color, texture, and SIFT, achieving a 61.34 percent classification rate for 50 different types of foods. Candidate zones are also discovered in [7], and testing findings reveal 56 percent accuracy when employing 10 classes of food types when using bag-of-words mixed with SIFT, spatial pyramid, histograms, and Gabor texture.

The authors of [4] offer a method that integrates a Deep Convolutional Neural Network with traditional hand-crafted image features, Fisher Vectors with HoG, and Color patches to greatly improve food recognition accuracy. The results demonstrate that for the 100-class food dataset, UEC-FOOD100, they scored 72.26 percent top-1 accuracy and 92.00 percent top-5 accuracy, outperforming the best classification accuracy of this dataset so far, 59.6 percent.

The effectiveness of the deep convolutional neural network used to food recognition was investigated by other writers, [8]. Their research looked for the ideal mix of large-scale ImageNet data pre-training, fine-tuning, and activation features extracted from the pre-trained deep convolutional neural network. The results suggest that employing a pre-trained network with 1000 food-related categories was the most accurate way, with UEC-FOOD100 attaining 78.77 percent accuracy and UECFOOD256 achieving 67.57 percent.

3 Experimental Setup

The UEC FOOD 256 is used as a use case in this study. There are around 32.000 photos in this dataset, spanning 256 different cuisine categories. A bounding box is attached with each food image, showing the food's position. The majority of the cuisine categories in this dataset are well-known in Japan. As a result, several categories may be unfamiliar to non-Japanese speakers. The dataset was separated into three sections: 60% for training, 20% for testing, and 20% for validation.

The research was carried out using a computer with an Intel i5 3.4GHz processor, 16GB RAM, an Asus Nvidia GeForce GTX 1070 8 GB graphics card, Windows 10 64-bits, and MatLab R2018a. The identical parameters were set for all CNNs evaluated (GoogleNet; Inception v3; ResNet 101), as shown in Table 1.

Table 1. Testing configuration for all CNNs

Training Cycle	
Epoch	6
Iterations:	15264
Iterations per epoch:	2544
Validation	
Frequency	3 iterations

A small test group of 20 participants was formed to compare the CNNs and the HVS. Ten were female, and the other ten were male, all between the ages of 25 and 40, and all of European descent. Europeans, like CNNs, are unfamiliar with Japanese cuisine. As a result, prior to testing, training is required. As a result, the investigation was split into two halves. The subjects were given a training stage that included a series of 98 slides for each of the 16 food groups. On each slide, seven photographs from the chosen food category are shown in random order, along with one image from other food categories. The test subject then had to choose the one that did not fall into the appropriate meal group. This permits the test subjects to learn the names of the food categories as well as their main qualities. In the second stage, picture classification, a collection of images from the 16 trained classes is displayed at random, each with 16 class label alternatives, of which only one is right. The test participants must choose the correct label for the image. There were 686 food photographs in the HVS collection, with 43 photos in each class.

3.1 Bag-of-Words architecture

A machine learning classification technique known as bag-of-words. The method converts photos into words and generates a histogram of the visual representation in the image. An image recognition classifier is built using histograms like

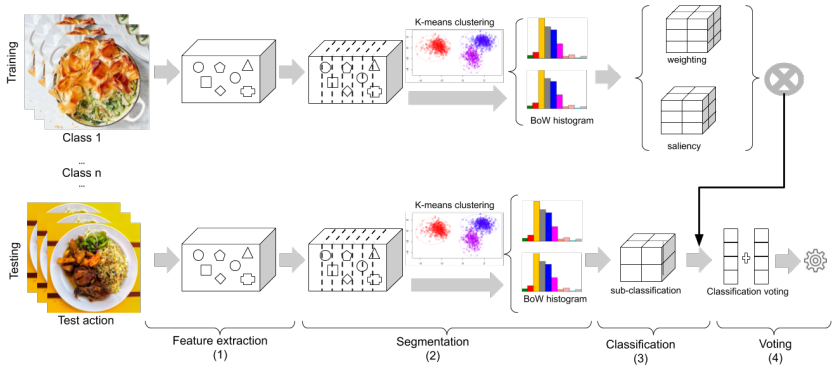

Fig. 1. Bag-of-words architecture sketch

these. In general, using a training dataset, the bag-of-words extract features to represent the image, generate a visual vocabulary (bag of features), and then classify the image.

The bag-of-words architectural processes to extract features and classify an image are depicted in Figure 1. The feature extraction for both training and testing pipelines is depicted in (1). Vectors of features are used to characterize these features in the bag-of-words technique. In this stage, features such as texture (GLCM binary patterns), color histograms, geometry properties of regions (6 layers at the start of step 2), and SURF were extracted, all of which were orientated to the proposed research.

Image features from the training dataset are extracted and clustered using a k-means technique, resulting in k feature vectors. The centroid of each class feature from the training dataset (2) is represented by the k feature vectors; visual words are grouped and segregated by similar qualities and characterized as a vocabulary histogram. A trained artificial neural network combines all weighted vectors and saliencies assessed using a ReLU function in the classification step (3).

The forward pass of the deep convolutional network, which only leaves the input's positive components, is the same as feature visualization. Following step (3), the previously trained neural network feeds the testing images and labels them in step (4). The approach finds and removes characteristics from the training photos for every new picture for classification and creates the histogram for the picture occurring in codewords once the BOW has been trained. The qualified classifier then assigns the image to one of the groups (step (4)).

3.2 Deep learning architecture(s)

A set of networks were tested to compare the accuracy of CNNs: GoogleNet, Inception-v3, and Resnet101. Prior to Resnet, neural networks had difficulty with gradients that were ignored and subsequently eliminated during the back-propagation learning process, reducing the number of layers. As a result, Resnet

architecture improves how Stochastic Gradient Descent is used in deep-learning-networks training via residual modules. Residual modules are sub-architectural blocks that feed the next layer as well as levels two and three hops away. Deep-learning networks can minimize vanishing gradients concerns and categorize more accurately by passing leftover modules forward rather than rejecting them.

Several layers compose Resnet, and at each layer, the output of the previews is added. Resnet follows VGG's full 3×3 convolutional layer design. The residual block has two 3×3 convolutional layers with the same number of output channels. Batch normalization and a ReLU activation function follow each convolutional layer. The input is then added directly before the final ReLU activation function using these two convolution procedures. The output of the two convolutional layers must have the same form as the input to be joined together in this configuration. Let us focus on a local neural network, as depicted. Denote the input by x. The ideal mapping by learning is $f(x)$, used as the activation function input. The portion within the dotted-line box must directly fit the mapping $f(x)$.

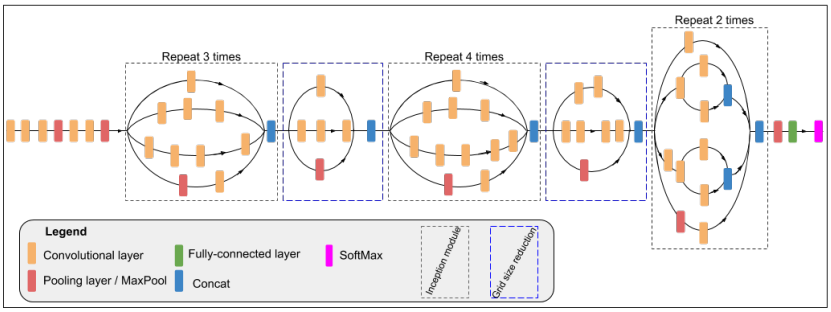

Fig. 2. Simplified inception architecture, ConvNets model

GoogleNet first introduces the inception architecture concept, and later the CNN inception improves it. Inceptionv3 CNN introduces a particular inception module to improve model performance, a multi-level feature extraction that computes 1x1, 3x3, and 5x5 convolutions, all in the same network module. Convolutional layers with varying filter widths are computed in parallel in the inception module. Before moving on to the next layer, the features are concatenated. The model's learning power is considerably increased as the number of features grows. In addition, because pooling operations are required by the Inception convolutional network, a parallel pooling path has been implemented to reduce the amount of data and processing time (grid size reduction). The probabilities of each class are output via Softmax activation. The architecture for the inceptionv3 model is described in Figure 2. Note that the outputs of the inception filters are all stacked and used as input to the next layer.

In the experimental testing setup, the number of layers for the architectures, Resnet, Inception, and Googlenet, were 101, 48, and 22, followed by training

for a more significant number of epochs (300) to ensure the convergence of the network. All the three CNNs were pre-trained (with a selected set of images) to classify images (from another distinct collection of images) from the food dataset UECFOOD256. The training and classification were either all 256 categories or 16 randomly selected categories. The training rate was initially configured to 0.05, validating every four interactions. Preliminary conclusion indicates accuracy convergence in every run. Procedures to classify new food images consisted of loading the food image to be categorized, extracting features automatically (using the CNN convolutions layers), and giving learned weights to each feature. As a result, food is classified as one of those types, ending with a group of probabilities that the submitted food is one of the before learned ones. Note that an independent dataset is used to evaluate accuracy.

3.3 Baseline survey

A survey of Asian dishes was conducted and sent to those who had no prior knowledge of these cuisines. A test group of 20 subjects was accepted (those who did not know at least 80% of the dishes), all of whom were between the ages of 25 and 40, and all of whom were European nationals. Two of the survey's control questions were universal foods, such as pizza.

Before testing individuals, it is required to undergo training. The task then becomes training the individual to correctly identify the names of food dishes given to him, with accuracy measured as the proportion of accurate choices. As a result, the survey was split into two parts.

The human subjects were given a training stage consisting of 98 screens for each of the 16 food categories (a number that was regarded modest enough to maintain humans' attention but not too little to be too easy). Each screen displays seven photographs from the selected food category as well as one image from a different cuisine category in a random arrangement. The test subject then had to choose the one that did not fall into the appropriate meal group. This helps the test takers to memorize the names and key attributes of each food type.

In the second stage, picture classification, a series of photos from the 16 learned food dish classes, each with 16 class label alternatives, is shown in random sequence on 32 displays, with only one right option. The test subjects must choose the appropriate label for the image.

4 Comparison study

Using the UEC Food 256 full dataset, the following CNNs were trained from scratch to classify different types of foods: GoogleNet, Inception v3, ResNet 101.

Table 2 shows a summary of the comparison of the obtained accuracy and training time results. When comparing all, ResNet 101 and GoogleNet converge faster (in fewer epochs) to the final result. In comparison, Inception V3 "climbs" to the final result/accuracy more gradually over the training time (more epochs

Table 2. Results comparison, validation accuracy, validation loss, and training time

	Results GoogleNet	**Results Inception v3**	**ResNet 101**
Validation accuracy:	47.73%	65.43%	70.68%
Validation loss:	2%	1.4%	1.2%
Training time:	6105 min. 12 sec.	5290 min. 42 sec.	7678 min. 17 sec.
Classification time (per image):	0.138 sec.	0.655 sec.	0.27 sec.

are necessary). On the other hand, Inception v3 shows more potential to improve the accuracy performance if extended training is performed. Nevertheless, ResNet 101 obtained better accuracy with the same test conditions.

From the confusion Matrix for CNN, resNet 101, with all 256 classes considered, most miss-classifications fell outside the 16 considered classes. With the CNN ResNet 101, with all food classes of the dataset available to classify, only the 16 classes used in the HVS survey inquiry, this CNN reached an accuracy of 67.7%. The survey inquiry performed to our test group, with humans, reached an accuracy of 80.6%, 13.4% more than the CNN.

Table 3. CNN vs. HVS results

	HVS	**CNN ResNet 101**
Validation accuracy:	80.6%	67.2%
Training time (avg):	17 min. 56 sec.	7678 min. 17 sec.
Classification time (avg):	14 sec. (per image)	0.27 sec. (per image)

Table 3, resumes the results for 16 food classes classification, with the knowledge of all 256 food classes, and 16 food classes classification of 16 food classes training, with a lifetime of food knowledge. When considering all trained food classes for both CNN and human test subjects, HVS performed better than CNN. However, we must consider that both make mistakes when:

- The training dataset does not comprehensively cover the classification images.
- When the image to classify is too different from the learned ones.
- Other dishes similarities influence the classification (i.e., overfitting).

The CNNs were previously taught with 256 food categories in prior tests. The three CNNs were retrained from scratch in the next experiment. They were just evaluating the HVS inquiry training images for training and categorization. The following subsection displays the results.

4.1 Additional results, CNN (16 classes) vs. HVS

Additional tests were conducted using the three CNNs in this subsection, utilizing only 16 food classes (and the same six epochs). With these results, it is feasible to determine which CNN is more efficient when trained in the same way as the HVS and how the results compare to the HVS in this situation. The three CNNs were trained from scratch using the same images used during HVS inquiry for training and classification. The images in the CNN training dataset were identical to those in the HVS training set, with a total of 686 food shots divided into 43 classes.

In Table 4, results of the CNNs training progress (accuracy and loss). Both GoogleNet and InceptionV3 reached the same accuracy (85.92%), and ResNet101 was the most efficient (89.89%). However, ResNet101 was the CNN that took more time to train, and GoogleNet the fastest. For comparison purposes, the most important conclusions are that:

- HVS errors are constant and visually/syntactically related.
- CNN's errors are much more dispersed and (for us humans) without visual relation.

Table 4. Results comparison, 16 food classes, 6 epoch: validation accuracy, validation loss, and training time

	Results GoogleNet	**Results Inception v3**	**ResNet 101**
Validation accuracy:	85.92%	85.92%	89.89%
Validation loss:	0.45%	0.5%	0.3%
Training time:	15 min. 18 sec.	73 min. 27 sec.	94 min. 26 sec.

4.2 More (20) epochs impact on CNN accuracy

The number of training epochs was then raised from 6 to 20. The dataset was identical to that used in the preceding and HVS trials (16 classes).

It is possible to conclude from the training process for the three CNNs using 20 epochs that the CNNs do not considerably enhance their accuracy beyond around ten epochs.

The accuracy, loss, and training times of the three CNNs are compared in Table 5. CNNs can enhance their accuracy greatly by utilizing 20 epochs instead of 6, almost stabilizing at a maximum of roughly 90% accuracy.

When comparing the CNNs with the HVS over a period of 20 epochs. CNN mistakes, like 6 epochs, are distributed throughout a wide range of dietary categories and are not immediately associated.

Table 5. Results comparison, 16 food classes, 20 epochs: validation accuracy, validation loss, and training time

	Results GoogleNet	**Results Inception v3**	**ResNet 101**
Validation accuracy:	89.17%	90.25%	93.86%
Validation loss:	0.30%	0.30%	0.25%
Training time:	49 min. 31 sec.	237 min. 22 sec.	288 min. 28 sec.

4.3 Human, Resnet, Inception and Bag-of-words

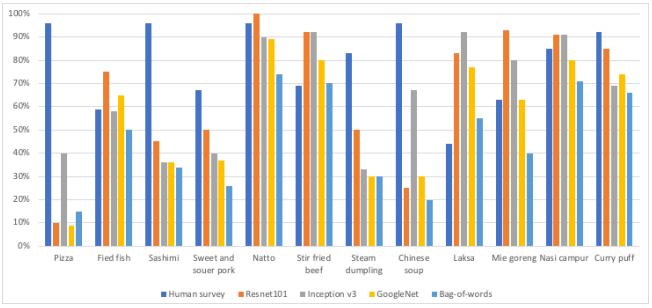

Fig. 3. Some food classification accuracy's

Resnet101 took 8018 minutes to train for 256 food categories and around 226 minutes to train for 16 classes. The accuracy of the methods Human, Resnet101, GoogleNet, Inception v3, Bag-of-words (500 words vocabulary), and Bag-of-words (1000 words vocabulary), all with two approaches, 16 food categories and 256 food categories, is compared in Figure refimg: Accuracy comparison (except for the human survey). In the bag-of-words, data demonstrate that increasing the number of codewords has little effect on accuracy, and the bag-of-words was still inferior to CNNs and humans. CNNs outperformed humans in 16 food categories, according to the findings. A more realistic test using 256 food categories shows that CNN's performance drops dramatically. Resnet was the best-performing CNN, with 95% and 75% accuracy for 16 and 256 food categories, respectively.

Figure 3 shows some class precision values from classification of: Humans (16 food classes); Resnet (256 categories); Inceptionv3 (256 categories); GoogleNet (256 categories); and Bag-of-words (1000 code-words). The analysis of the values in Figure 3 allow concluding that the hardest classification categories for humans and neural networks are different, which lead to humans and CNNs having distinct behaviors when classifying types of food types.

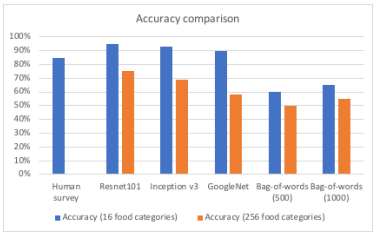

Fig. 4. Accuracy comparison.

A global analysis of results (Figure 4 and 3) show that CNNs are more accurate than bag-of-words, simultaneously show that as the number of categories to learn increases (16 to 256 food categories), the accuracy decreases. Consequently, human knowledge continues to outperform the accuracy of CNNs when identifying different food types.

5 Conclusions

When comparing the CNN to the HVS for a limited number of classes (16), the HVS outperformed the CNN, with 67.2 percent CNN accuracy vs. 80.6 percent HVS. When comparing the HVS to CNNs trained with only 16 classes, the CNNs' accuracy was much higher than the HVS, reaching 89.89 percent. Additional testing demonstrate that using only 16 food classes and increasing the number of epochs from 6 to 20 improves accuracy even more, reaching 93.86 percent. When it comes to categorization, the main distinction between CNN and HVS is that CNNs make errors with a variety of food classes, but HVS virtually always makes semantic errors with the same food classes. Therefore, HVS shows more consistency with the provided answers.

Humans and CNNs are compared to the bag-of-words. The experimental comparison procedure enables us to conclude that CNNs are more accurate than bag-of-words, while also demonstrating that CNN accuracy declines as the number of food categories grows (considering 16 to 256 food groups). As a result, when it comes to different types of food, human intelligence tends to outperform CNNs. Although different classifications of people and deep learning are robust, human and CNN's attitudes are distinct when classifying various food types.

Acknowledgements

"This work is funded by National Funds through the FCT Foundation for Science and Technology, IP, within the project Ref UIDB/05583/2020. Furthermore, we would like to thank the Research Centre in Digital Services (CISeD), the Polytechnic of Viseu, for their support."

References

1. N. Abou Baker, N. Zengeler, and U. Handmann. A transfer learning evaluation of deep neural networks for image classification. *Machine Learning and Knowledge Extraction*, 4(1):22–41, 2022.
2. F. Chen, J. Wei, B. Xue, and M. Zhang. Feature fusion and kernel selective in inception-v4 network. *Applied Soft Computing*, page 108582, 2022.
3. R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.
4. R. Khan, S. Kumar, N. Dhingra, and N. Bhati. The use of different image recognition techniques in food safety: A study. *Journal of Food Quality*, 2021, 2021.
5. G. Lagani, F. Falchi, C. Gennaro, and G. Amato. Comparing the performance of hebbian against backpropagation learning using convolutional neural networks. *Neural Computing and Applications*, pages 1–17, 2022.
6. M. V. Lande and S. Ridhorkar. A comprehensive survey on content-based image retrieval using machine learning. *Proceedings of Data Analytics and Management*, pages 165–179, 2022.
7. R. Mao, J. He, Z. Shao, S. K. Yarlagadda, and F. Zhu. Visual aware hierarchy based food recognition. In *International Conference on Pattern Recognition*, pages 571–598. Springer, 2021.
8. K. Ohri and M. Kumar. Review on self-supervised image recognition using deep neural networks. *Knowledge-Based Systems*, 224:107090, 2021.
9. N. O. Salim, S. R. Zeebaree, M. A. Sadeeq, A. Radie, H. M. Shukur, and Z. N. Rashid. Study for food recognition system using deep learning. In *Journal of Physics: Conference Series*, volume 1963, page 012014. IOP Publishing, 2021.
10. P. Sharma, A. SHARMA, et al. Hybrid approach for food recognition using various filters. *International Journal of Advanced Computer Technology*, 11(1):1–5, 2022.
11. H. Touvron, M. Cord, A. Sablayrolles, G. Synnaeve, and H. Jégou. Going deeper with image transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 32–42, 2021.
12. A. M. Zahangir, H. Mahmudul, Y. Chris, T. M. Taha, and V. K. Asari. Inception recurrent convolutional neural network for object recognition. *Machine Vision and Applications*, 32(1), 2021.
13. Y. Zhu, R. Urtasun, R. Salakhutdinov, and S. Fidler. segdeepm: Exploiting segmentation and context in deep neural networks for object detection. In *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on*, pages 4703–4711. IEEE, 2015.