Nutritional Value Detection Using Computer Vision

James Beckett
Institute of Information & Communication Technology
Malta College or Arts, Science & Technology
Corradino Hill
Paola PLA 9032
james.beckett.d56942@mcast.edu.mt

***Index Terms*—MCAST, IICT, Computer Vision, YOLOv5, Python, C#**

I. Chosen Research

Being aware of one's nutritional intake is a important factor in looking after physical and mental well-being, this paper aims to investigate whether Computer Vision can be used in order to aid in streamlining the process of collecting Nutritional Information from cooked meals made with fresh ingredients such as vegetables, which may not normally be as simple to obtain the nutritional value of due to a lack of packaging displaying this information to the consumer.

The rationale for choosing this topic is that there is a lack of solid research in to whether Computer Vision is truly a feasible alternative to carrying out the practice of obtaining Nutritional Information manually or whether it is ineffective as a replacement for this method. There has been a great deal of research in to the effectiveness of computer vision algorithms to detect ingredients and even multi-ingredient plates, however there has been very little research carried out with regards to how effective these results can be in a practical setting such as for nutritional information gathering. This research aims to clarify whether these algorithms can detect food effectively enough to provide an accurate estimate of nutritional value to a user.

The research philosophy adhered to throughout is this paper is that of positivism. This philosophy was chosen as the main goal for this paper is to clearly define using objective data and analysis the effectiveness of a computer vision object detection model in estimating nutritional information. This will be done using deductive reasoning to test the hypothesis presented and come to an objective conclusion on just how effective this method can be by developing a prototype and obtaining results outputted by running it.

The case-study research strategy is the most appropriate strategy to make use of in this instance as a thorough, detailed study of Computer Vision algorithms and their strengths and limitations is required in order to gain a strong understanding of how these systems work. Once this is complete results will be obtained through prototype testing, meaning this case-study is entirely focused on obtaining quantitative data. This is a mono-method approach as research questions were outlined and then data collection was carried out before being analysed and later interpreted to come to a conclusion at the end of research.

Data for this study was gathered in a cross-sectional fashion, meaning that the research questions posed were focused around a specific version of an algorithm, in this instance YOLOv5 and the data was collected on-demand with no regard for time or need to collect data over a period of time. The main technique used in this study was statistcal analysis based on a high volume of data obtained through prototype testing.

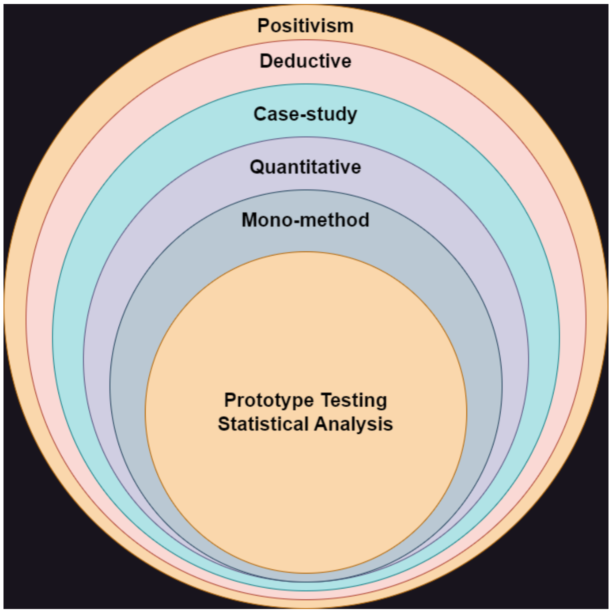

Fig. 1. The research onion

The points mentioned above lead to the hypothesis that it is possible to obtain accurate nutrition information of a cooked meal using a computer vision algorithm. The purpose of this study is to test this hypothesis and come to a conclusion as to whether it is true or false. The independent variable in this study are the results output by the prototype. The dependent variables are the dataset used for training the Object Detection model, the API used for nutrition information collect and the YOLOv5 framework itself.

II. Review of research methodology

A. Methodologies of Other Studies in the Field

In 2019 Benjdira et al. investigated the difference in performance between Faster R-CNN and YOLOv3 when preforming car detection. To perform this experiment, a dataset was created that was split in to a training set and a testing set, the training set containing 218 images and 3,365 instances of cars and the testing set containing 52 images and 737 images of cars. With regards to Faster R-CNN, the Tensorflow Object Detection API was used with the Inception ResNet v2 CNN network for training. Training was optimised using stochastic gradient descent with momentum set to 0.89 and the learning rate set to 0.00019. The training was then taken out for 200,000 steps on images with a minimal dimension of 600 and a maximum dimension of 1024. A random horizontal flip was used for data augmentation and the batch size was set to one. The algorithm was trained only to recognise one class which is the class ”car”. [1] The training of YOLOv3 was taken out using the YOLO provided code with default configuration as the parameters. Training was optimised using stochastic gradient descent with momentum set to 0.9, the learning rate set to 0.001 and the weight decay set to 0.005. The values of height and width were both set to 608 and batch size was set to 64. The specifications of the computer used for both of these configurations used an Intel Core i9 CPU in tandem with an NVIDIA GTX1080 8GB GDDR5 Graphics Card and 32GB of RAM [1].

TABLE I
Evaluation Metrics of Faster R-CNN and YOLOv3

Measure	**Faster R-CNN**	**YOLOv3**
True positives	578	751
False positives	2	2
False negatives	150	7
Precision (TPR)	99.66%	99.73%
Sensitivity (Recall)	79.40%	99.07%
F1 Score	88.38%	99.94%
Quality	79.17%	98.81%
Processing time (Avg. in ms)	1.39s	0.057ms

Table I and its results indicate that both algorithms have a high precision rate, meaning that when they classify an object as a car they are likely correct. However, it is clear that YOLOv3 has a significantly better recall than Faster R-CNN outperforming it by 19.67%, indicating that YOLO is considerably better at finding all cars present in an image than Faster R-CNN which is more likely to miss cars (false negatives). In addition to this YOLOv3 is clearly considerably faster than Faster R-CNN with an average processing time of 0.057ms, faster than Faster R-CNN’s 1.39s by 1.33 seconds. In conclusion, it is evident that YOLOv3 vastly exceeds Faster R-CNN’s performance in both accuracy and speed and therefore it was determined that for Benjdira et al.’s needs it is the *”best algorithm for object detection”* [1].

A 2017 study by Takumi Ege and Keiji Yanai focused on using R-CNN to detect food dishes, specifically from a dataset of common Japanese school lunch dishes. This dataset consisted of 4877 images all annotated with bounding boxes, the labels used for these bounding boxes were a combination of specific dishes such as bibimbap and fried noodles, as well as more broad categories such as meat and fish. In this study, Faster R-CNN was used to detect the dishes. The algorithm was training using an implementation released by the author, with a momentum of 0.9 and a weight decay of 0.0005. Once training was complete the model was then evaluated using mean Average Precision (mAP). 80% of food photos in the the built dataset were used for training, with the other 20% used for performance evaluation. The mAP of the model as a whole was 90.7%, indicative of an accurate model on the whole [2].

TABLE II
Average Precision of Faster R-CNN Model detecting School Lunch

Food	**Average Precision**
Milk	99.6%
Drinkable yogurt	90.6%
Rice	99.7%
Mixed rice	82.7%
Bread	95.5%
White bread	83.7%
Udon	98.0%
Fish	78.3%
Meat	70.8%
Salad	94.0%
Cherry tomatoes	100.0%
Soups	92.2%
Curry	95.1%
Spicy chili-flavoured tofu	99.8%
Bibimbap	72.9%
Fried noodles	79.9%
Spaghetti	90.7%
Citrus	99.6%
Apple	98.5%
Cup desserts	93.1%
Other foods	90.4%
mean Average Precision	90.7%

Table II indicates that the model excelled when detecting food that are served individually as opposed to as a dish, that have distinctive, regular, clearly-defined shapes, such as cherry tomatoes (100%), citrus (99.6%) and apples (98.5%). In contrast, the model struggled with complex dishes made up of many ingredients such as bibimbap (72.9%) and fried noodles (79.9%), as well as struggling with the vague categories such as meat (70.8%) and fish (78.3%) [2].

B. Distinguish Between Academic and Non-Academic Material

The key distinctions to be made between academic and non-academic materials lie in their purpose and their credibility. Academic materials are normally peer-reviews, scholarly articles, journals or book excerpts, intended to contribute to a new or existing body of research. They are written by experts and often reviewed rigorously by other experts before publication. Due to this, academic sources provide detailed analysis of research cases, in-depth findings and theories to support arguments made. In contrast, non-academic materials can come come from a range of any number of sources including but not limited to, websites, magazines, newspapers and personal blogs. They can be useful to offer insight in to certain subjects, however they are aimed at a general audience and thus not subject to the same level of academic scrutiny as their counterparts. This means that it is vital to critically evaluate non-academic sources in order to determine their validity.

C. Recommended Articles

5 recommended articles from peer-reviewed journals are:

- *Food Calorie and Nutrition Analysis System based on Mask R-CNN* by Meng-Lin Chian, Chia-An Wu, Jian-Kai Feng, Chiung-yao Fang and Sei-Wanh Chen from pages 1721 to 1728 of the *2019 IEEE 5th International Conference on Computer and Communications (ICCC).*
- *Estimating Food Calories for Multiple-Dish Food Photos* by K. Yanai and T. Ege, from pages 646 to 651 of the *2017 4th IAPR Asian Conference on Pattern Recognition (ACPR)*
- *Car Detection using Unmanned Aerial Vehicles: Comparison between Faster R-CNN and YOLOv3* by Bilel Benjdira, Taha Khursheed, Anis Koubaa, Adel Ammar and Kais Ouni from pages 1 to 6 of the *2019 1st International Conference on Unmanned Vehicle Systems-Oman (UVS)*
- *Computer Vision: Evolution and Promise* by T.S. Huang from pages 21-25 of the *1996 CERN School of Computing*
- *Object identification for computer vision using image segmentation* by Debalina Barik and Manik Mondal from pages 170-172 of the *2010 2nd International Conference on Education Technology and Computer*

D. Literature Map

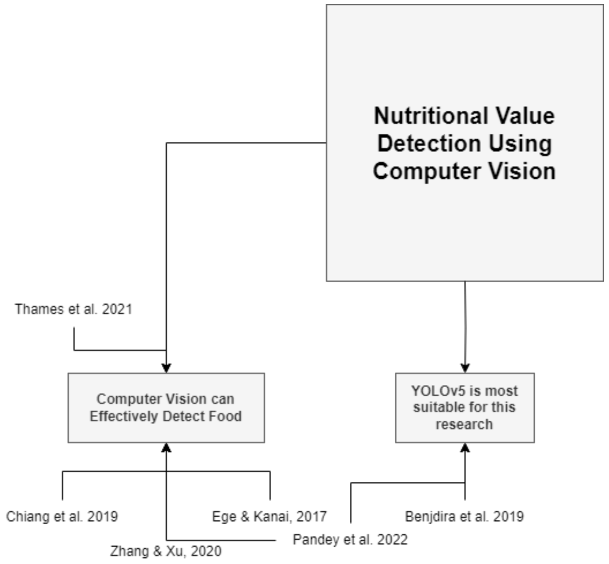
Nutritional Value Detection Using Computer Vision
Thames et al. 2021
Computer Vision can Effectively Detect Food
YOLOv5 is most suitable for this research
Chiang et al. 2019
Ege & Kanai, 2017
Benjdira et al. 2019
Pandey et al. 2022
Zhang & Xu, 2020

Fig. 2. Literature Map

III. Reflection on the Chosen Methodology

A. Defining Research Questions

Can a computer vision mode accurately distinguish between different types of food?, this question is set out to determine if a computer model, given a varied set of classes of ingredients, can accurately differentiate between them in a reliable enough way to be useful in other applications, such as gathering nutrition information.

What dataset is best for training an algorithm for this task?, the aim of this question is to determine which of the many computer vision algorithms available is most suited to the task of recognising multiple food objects on a plate, taking in to account accuracy, ease of training and efficiency.

What training settings are best for training an algorithm for this task?, is an important question as the settings used to train a computer vision model can drastically effect its effectiveness. There are variety of options controlling everything from input image size to training epochs (iterations).

B. Defining Objectives

In order to address the research questions above, the following will need to be carried out. A dataset will need to be obtained via downloading and manually labelling thousands of images, this dataset must then be used to train a computer vision model with the chosen settings in order to create an object detection model capable of recognising a variety of ingredients with a solid degree of accuracy. Once this has been completed, images of meals with known nutritional information will need to be gathered and detected using the trained model, as well as pictures of individiual ingredients for more detailed results. Finally, a CSharp program must be developed in order to accept manually-inputted detection results and return nutritional information from an API call to an appropriate API.

C. Research Philosophies. Approaches and Paradigms

With regards to research philosophies, the main three philosophies that one might use during their research are Positivism, Interpretivism and Pragmatism.

Positivism is a philosophy that emphasises objective measurement and direct observation of results. Being a positivist means believe that research should not adhere to any values or have any pre-conceived biases, but must instead focus on objectively determining what the real and true results of carrying something out a certain way are. To a positivist, if something is not true or false it is essentially meaningless. This means that abstract questions (such as those about existence of god) are irrelevant to positivists as it is impossible to obtain an obejctively correct answer to the question.

Interpretivism is essentially the opposite of positivism, in that interpretivists believe that the world and it's rules should be observed in a subjective light. This means that they lend a lot more value to social and cultural factors when carrying out research, focusing more on the participants in research than more objective results. They look to understand the perspective of participants and thus make greater use of qualitative data than positivists as it is more suited to explore complex social phenomena.

Finally, pragmatism is an approach that believes in practicality and using a variety of methods when conducting research. This approach acknowledges that the multiple research questions of a study may each require different techniques to be answered and thus makes use of multiple methods in order to answer each question as best as possible. Due to this, research can be carried out in a much more flexible manner, allowing for a combination of qualitative and quantitative methods in order to obtain the best results. Emphasising the importance of the practicality of research findings and how they may contribute to improving specific scenarios.

As for approaches, the two main approaches one might take are deductive reasoning and inductive reasoning. They key difference between both of these approaches is that deductive reasoning begins with a theory or hypothesis and then works towards testing out that theory through research, whether that be a prototype, survey or combination of both. In contrast inductive reasoning is the process of instead generating a theory based on research. Essentially, deductive reasoning begins with an assumption of what is going to happen, then research is carried out to determine whether that is the case or not. Whereas with inductive reasoning, research is carried out with the scope of figuring out what happens after it is completed, there is no main hypothesis to prove right or wrong but instead simply conclusions to come to based on research.

D. Chosen Methodology

Based on the methodologies reviewed it was determined that the correct course fo action for carrying out this research would be to use the YOLOv5 algorithm to train an object detection model capable of recognising ingredients. This algorithm was chosen as it presents by far the best results out of any of the viable options.

Once research is carried out as outlined above, compiling results will simply be a case of obtaining the detections from the model and aggregating the results, using these to analyse the model on an ingredient-by-ingredient basis as well as evaluating the accuracy when detecting the nutritional value of an entire meal. From here it will be possible to recognise the advantages of making use of such a model whilst also recognising its pitfalls and how it may be improved in future studies. There are no ethical considerations to take in to account throughout this research as at no point does it interact with people.

TABLE III
CONFIDENCE AND DETECTION COUNT BY INGREDIENT

Ingredient	Average Confidence	Detection Count
Steak	0.78	4
Minced beef	0.66	6
Chicken breast	0.72	13
Salmon	0.74	10
Bacon	0.62	8
Sausage	0.81	22
Fried egg	0.84	10
Boiled egg	0.79	41
Kidney beans	0.74	5
Peas	0.76	3
Peanuts	0.65	4
Cashews	0.65	9
Cheddar	0.78	10
Mozzarella	0.86	12
White bread	0.75	17
Wholemeal bread	0.7	3
Boiled potato	0.75	16
Roast potato	0.69	48
Fries	0.68	7
Pasta penne	0.78	4
Pasta spaghetti	0.64	4
Rice	0.7	8
Carrot	0.69	7
Broccoli	0.69	46
Tomato	0.78	13
Cucumber	0.76	19
Mushroom	0.69	19
Apple	0.76	21
Orange	0.85	12
Banana	0.78	22
Euro coin	0.9	25

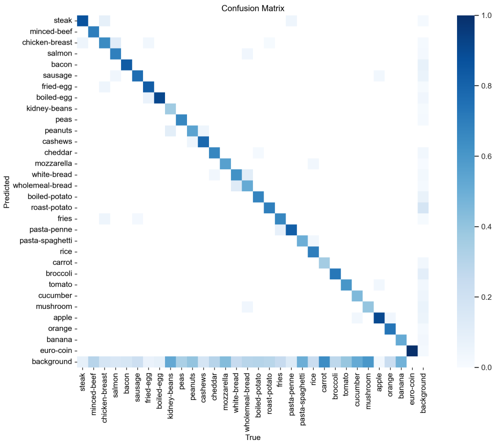

Fig. 3. Confusion Matrix for YOLOv5 Model

IV. Results, Analysis and Discussion

A. Average Confidence by Ingredient

In this section, the object detection model is analysed by looking at the average confidence of each detection of an ingredient, regardless of whether the detection was a true or false positive. This means that even if the model incorrectly detects one ingredient as another, it will still count towards the average confidence value. In addition to this, for added context, a count of how many detections were made by the model per ingredient was logged. This was done to give a better idea of why certain values may be low or high.

As seen in Table III, the highest confidence value of an ingredient belonged to mozzarella cheese, with an average confidence value of 0.86 from 12 detections. This means that when the model reported that it had detected mozzarella cheese, it was always very confident in its answer, this is likely due to the uniformity of this ingredient meaning that no two images of mozzarella look particularly different, causing less confusion for the model. In contrast, the lowest average confidence value reported by the model belonged to bacon, with an average confidence value of 0.62 from 8 detections. This is likely due to a search for the term bacon providing images of bacon in various shapes and sizes, leading to the model struggling to create a solid definition for what bacon looks like. One way to improve the confidence for bacon could be to split it in to two categories, *streaky bacon* and *back bacon*. This would allow the model to differentiate between both types more easily and give stronger detection results for both types of bacon.

Another observation that can be made from this data is that both wholemeal bread and peas possess the lowest count of detections out of all ingredients. However, it is likely that in each instance this is for a different reason. In the case of wholemeal bread, it is likely that the low number of detections correlates directly with a high number of false positives for white bread. This is due to both these ingredients being very visually similar when analysed in a two-dimensional space by a computer vision model. The two ingredients having similar confidence values lends to this theory.

With regards to the low detection count for peas however, this can likely be put down to peas being small objects normally served in large quantities, meaning that for the model to accurately detect a larger serving of peas, it must accurately detect a collection of small ingredients as one large group. Based on the findings in Table III this appears to be a common theme, as other ingredients that are often served as a group as opposed to individually such as rice, penne pasta and spaghetti all achieved less than 10 detections.

B. Results in Relation to Hypothesis

Whilst a lot more results need to be obtained, the above results show the early indications that the model trained will be useful in accurately detecting individual ingredients on a plate and combining those detections in order to provide an accurate nutritional evaluation. In spite of this, compared to other studies confidence and accuracy are a little lower than expected.

V. Conclusion

In this study it was determined that the YOLOv5 model is effective for identifying seperate food objects on the same plate and returning those results to the user to be used for nutritional information detection.

Addressing the research questions, it was determined that a computer vision model can accurately distinguish between different types of food, as with a few small exceptions due to ingredients looking very similar, the model reliably gave satisfactory results. It was also determined that the best dataset for training an algorithm for this task is YOLOv5, this conclusion was reached after the literature review and confirmed by the results obtained from using it. Finally, the best training settings for an algorithm to carry out this task were to use an image size of 640 pixels by 640 pixels and a batch size (number of samples processed before the model is updated) of 16 for 100 epochs, this however comes with the caveat that a batch size of 32 would provide better results if a machine capable of training using those settings was available. However, with the tools at hand 16 was deemed to be satisfactory.

A key shortcoming from the methodology is that an even larger dataset needs to be used to get the best possible results. Whilst a 3100 image dataset is satisfactory for 31 classes, the results would improve considerably by doubling or even tripling the amount of images in the dataset.

In further research in addition to the larger dataset, it would be a good idea to develop an app that automatically interprets the results output by the object detection model and returns nutritional information to the user, this was originally intended to be done using an android app however complications during development and time constraints meant that this could not be carried out.

References

[1] B. Benjdira, T. Khursheed, A. Koubaa, A. Ammar, and K. Ouni, “Car detection using unmanned aerial vehicles: Comparison between faster r-cnn and yolov3,” *2019 1st International Conference on Unmanned Vehicle Systems-Oman (UVS)*, pp. 1–6, 2019.

[2] T. Ege and K. Yanai, “Estimating food calories for multiple-dish food photos,” *2017 4th IAPR Asian Conference on Pattern Recognition (ACPR)*, pp. 646–651, 2017.