Injecting prior frequency information in DETR for wheat head detection

Alin Prundeanu[1] ,
Yancong Lin[1] , **Attila Lengyel**[1] , **Silvia Pintea**[1]
[1]TU Delft
A.Prundeanu@student.tudelft.nl,

Abstract

Wheat is among the most important grains worldwide. For the assessment of wheat fields, image detection of spikes atop the plant containing grain is used. Previous work in deep learning for precision agriculture employs the already established object detectors, Faster R-CNN and YOLO, adapted for the given context. However, these models suffer from the necessary duplicate-removal post-processing and from the low performance on overlapping objects. On the other hand, the novel Detection Transformer (DETR) object detection approach manages to overcome such limitations, being an end-to-end anchor-free set predictor based on the transformer architecture, using the attention mechanism for modelling long-range dependencies. Consequently, the general sensitivity of this technique for small size objects in the wheat head domain is reduced. Nonetheless, previous research reflects the potential of frequency analysis techniques to increase the accuracy of a CNN. This paper aims to study the feasibility of adding frequency information as a pre-processing step to improve the performance of the DETR model for wheat head detection. Two variants of the original DETR with a mask based on the Fast Fourier Transform (FFT) of the power spectra of wheat heads and background patches are proposed and explored for improvements in prediction quality. Although promising, the best FFT-based DETR approach manages to deliver an average score of only 0.42, a slightly sub-optimal performance compared to DETR's one of 0.47. Additionally, as to grasp a sense of their capability among well-established detectors, YOLO-V3 and Faster R-CNN manage to achieve around 0.7 on the same wheat data set. Ultimately, a configurable automated overview of the development of wheat fields leads to a more efficient administration of the production process. To such end, this research explores the possible application of this new object detector in precision agriculture and provides insight into its limitations and potential ways of overcoming them.

1 Introduction

The significant importance of wheat worldwide, based on grain acreage and the total production volume [16], makes it widely studied. To gather large and accurate data about wheat fields, image detection of wheat heads [1] is utilized to estimate their density and size in different varieties. The data is used by farmers to assess health and maturity when making management decisions in their fields. In order to identify the wheat heads for further assessment, deep learning detectors are used on the images. An improvement in the quality of these estimations could enhance the wheat production, by having a more comprehensive and scalable overview of the development of wheat fields and therefore a more efficient administration of the production process.

A novel technique for object detection, the DETR approach [4], could prove as an interesting tool for accomplishing the wheat head object identification. It is an end-to-end, anchor-free, set prediction model based on a Convolutional Neural Network (CNN), a transformer encoder/decoder and a predictive Feed-Forward Network (FFN). Being end-to-end, it avoids the need of further conditioning the learning process on additional hand-crafted intermediate parameters. Here, the ability for anchor-free detection provides great flexibility for the identification of a variable group of objects that is not dependent on a predefined number of chosen anchors. Combined with the incorporation of the transformer logic, which presents itself particularly well suited for the bounding box detection task [2], the result is that more overlapping objects can be accurately identified. Additionally, DETR manages to bypass the post-processing step, employed by many state-of-the-art detectors, by using a direct set prediction approach. It relies on a minimum bipartite matching loss between the set of output predictions and the set of ground truth labels. This loss is a total function that is also permutation invariant, mapping elements of those two sets (predicted VS actual) in such a way that the sum of pairwise losses is minimized. As such, no duplicate predictions will be present in the final output and

[1] spikes atop the plant containing grain

[2] Since the attention mechanism is able to extract and aggregate the image features in a many-to-many relationship, certain distant elements which reveal the presence of a specific object or class can be identified by the attention layers, even though most of the object is occluded.

the need for the duplicate removal extra step is eliminated, leading to a simplified faster algorithm. Thus, the presented robustness of this network justifies its choice as the underlying model on top of which this study will focus.

On the other hand, two issues arise from placing the DETR model in the context of wheat detection. First, the traditional pipeline approach enabled some prior information, extracted by different intermediate steps, to be injected into the algorithm in order to improve its performance. Since the end-to-end philosophy no longer has this prior information available, it usually comes at a cost of requiring a much larger training dataset in order to achieve comparable results. Unfortunately, the available wheat head data set provided for this research is small, risking to be an insufficient supply for a complete learning process. Another crucial aspect pertains to the model's sensitivity for the size of the detected objects. More specifically, its average precision on small objects is proportionally lower than the one for medium and large size objects. Additional optimizations like "adding a dilation to the last stage of the backbone and removing a stride from the first convolution of this stage" [4] also involve an increased computational cost. Therefore, those two problems counterbalance the model's strong features of anchor-free detection with set prediction and attention mechanisms. As such, the premise of this research relies on addressing these expected limitations in performance.

1.1 Contribution of this research

The potential solution to the aforementioned obstacles, small dataset and small sized target objects, lies in manipulating additional frequency information before the detection process. Transforming every input image based on a frequency filter that specifically targets the wheat head properties has the potential of facilitating the learning process. It could improve performance by having less insignificant background information channeled into the model's internal state. Nevertheless, delivering more relevant data could compensate for the reduced training amount of it, overcoming one of DETR's limitations.

The previously mentioned aspects as well as the related work described in Background Section 2 reflect the potential of frequency analysis techniques to filter the input dataset in order to increase the percentage of relevant information that is being fed into the CNN machinery and therefore, the CNN's accuracy. Combined with the DETR framework, which builds upon an initial CNN feature extractor, it presents an interesting assumption: *Filtering the input images based on wheat head FFT mask improves the performance of DETR for wheat head detection.* Ultimately, the contribution of this research is to explore and assess the validity of this assumption. Meanwhile, investigating this supposition naturally leads to the following sub-questions, all placed in the context of wheat head prediction:

- Is DETR a suitable model for predictions in this domain?
- Does a filtering of the input based on frequency analysis improve the performance of the DETR predictor?
- How significant is the performance difference between the simple DETR and the FFT-based one? Why?
- What does the DETR performance mean when compared to the results achieved by other models that were used in the competition (YOLO, Faster R-CNN)? Taking into consideration that the models participating in the competition have been specifically tuned for better accuracy.

2 Background

2.1 Object detectors in precision agriculture

Part of the associated research performed in the domain of precision agriculture with deep learning is offered by the Global Wheat Detection [1] Kaggle competition, which is also the starting point of this study. Here, special emphasis is put on developing a model capable of a strong generalization, given the difficult characteristics of the presented images: overlap of wheat plants, wind blurring the photographs, different appearances of wheat plants (due to genotype) and wheat fields (due to growing environment). Just as observed in Kaggle, current detection methods involve one and two-stage detectors, Yolo-V3 and Faster R-CNN, despite for them incurring a bias to the training region, even when trained with a large dataset. Another work in the field of precision agriculture involves a transfer learning ConvNet for crop growth stage estimation [15]. Additionally, [14] presents a deep learning approach to detect flowers in soybean fields, by comparing the behavior of three well-established object detection methods: RetinaNet, Faster R-CNN, and Cascade R-CNN. However, most of these models suffer from the necessary multi-stage processing and from the low performance exhibited on overlapping objects. The post-processing of near-duplicate predictions, most often with Non-Maximum Supression, is a crucial step for their accuracy and the dependency on its manually set threshold value makes those methods struggle to generalize. Since the DETR approach manages to surpass this heuristic constraint, it has a better generalization capability across domains, encouraging transfer learning with faster training times for achieving reasonable results. In our case, this means that DETR pretrained models are a viable option on the Wheat Dataset.

2.2 DETR

The DETR's transformer network was originally introduced for the Natural Language Processing (NLP) tasks, building on top of the artificial attention schema pioneered by Vaswani et al. [19]. This attention mechanism enabled whole synchronous processing of the sequential data, improving the ability to model long range dependencies between input elements, focusing on correlations between possibly distant words in a sentence or pixels in an image. Furthermore, DETR has already been successfully employed, although slightly modified from the first version, in some object detection tasks, like identifying crowd pedestrians [10] and road accidents [17]. These methods highlight the state-of-the-art capability of the network either when coupled in a multi-stage Machine Learning (ML) pipeline or once some improvements in the set prediction loss component and the transformer layer are exposed and addressed.

However, this model has not yet been studied in a precision agriculture task. Its previously described versatility proposes the challenge of testing this framework on the Global Wheat Head dataset. Some basic DETR models have already been applied on the Kaggle dataset as part of the competition, but their performance remains relatively modest compared to YOLO and Faster R-CNN. Apart from [10; 17], this study does not pursue changes in the underlying architecture of the DETR in order to improve its behaviour and mostly relies on the proposed frequency domain prior.

2.3 Fourier analysis in deep learning

Current work in the area of enhancing a deep network with domain knowledge explores various ways to use frequency information in order to achieve better results. In [5] it is shown that by using a CNN pre-processing layer based on the unsharp masking algorithm, the end-to-end paradigm is kept while prior high-frequency information is also added to the input and injected in the network, resulting in an improved accuracy. Also, [7] and [12] both propose a similar CNN based on the Fast Fourier Transform (FFT), with a reduced complexity and faster training times. It relies on the application of FFT, which allows the convolution operation to be converted into multiplication, therefore having all the computation performed in the Fourier domain. Conversely, [6] highlights a variant of a CNN architecture with better precision and learning time, "which exploits the inherent redundancy in both convolutional layers and fully-connected layers of a deep learning model". Inspired by the proposed methods, this study also seeks a functional integration between an additional convolutional layer and the extracted frequency information. However, the end-to-end training that is preserved in [12; 6] is now dropped in favor of the extra FFT-based filtering step.

3 Methodology

In order to provide a concrete analysis of the proposed hypothesis, the methodological approach has been split according to the basic building elements. Conceptually, one part of the subject involves the DETR model and a brief exploration of its configuration for the Wheat dataset. The other part targets the frequency filtering prior enhancement that is added to the configured network and its effects on it.

3.1 The model

This study is built upon the DETR framework as it was originally presented in [4]. As a first challenge, the proper configuration of the network has to be found, such as to allow a learning process in the Wheat Head domain. The goal does not seek a competitive performance compared to other models used in the Kaggle challenge. It only targets a reasonable enough performance that allows solid conclusions to be drawn, so that proper examination of the network's behavior is facilitated.

Pertaining to its architecture, the model starts with a CNN backbone, as it can be observed in Figure 1. This is an ImageNet pre-trained ResNet50 or Resnet101 [9] network, that acts as an image features extractor. Next, the spatial dimensions of the resulting feature map are collapsed, resulting in a

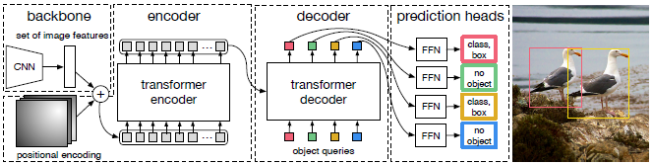

Figure 1: The DETR model.
Carion et al (2020). End-to-End Object Detection with Transformer.
Source: https://arxiv.org/abs/2005.12872

sequential flattening of those features ready to be injected into the transformer stage. Next, an encoder and a decoder focus on transforming N embeddings of size d using multi-headed self-attention (both) and encoder-decoder attention mechanisms (only decoder). Since the transformer architecture is permutation-invariant, positional encodings are added to the input of each attention layer in order to keep spatial information otherwise lost. This way, each feature extracted by the CNN is mapped to its corresponding region of the input image and plugged into the encoder. Then the decoder takes as input the encoder's output and also a small fixed number of learned positional embeddings, referred to as object queries. Those object queries are similar to the encoder's positional representations, only that they now dynamically map the output instead of input, by localizing the attention mechanisms. They are independently interpreted by a predictive feed forward network into either box coordinates and class labels or a "no object" class. This setup promotes attention mechanisms to focus on different regions of the input, as "the model globally reasons about all objects together using pair-wise relations between them, while being able to use the whole image as the context" [4].

The original DETR network has been trained on the COCO dataset [11], which provides a robust source of training images with different properties, classes and inter- as well as intra-class differences. By comparison, the Wheat Head Dataset only contains 1% of the COCO's size, with some of the images even presenting inaccurate bounding box annotations. Therefore, all the models used in this research use transfer learning built on top of the COCO [11] pre-trained DETR weights. Considering the diversity of the COCO dataset, the DETR's CNN pre-trained backbone that was exposed to it must have learnt effective kernels for image structure discrimination. Since the wheat head features have no similar analogue in COCO, most of those pre-trained kernels have been retained and generalized on the Wheat Head domain. Additionally, the already trained positional embeddings have managed to effectively assign attention maps that cover the entire image, regardless of its exact content. Those arguments already provide a higher starting ground for the pre-trained network's performance in our wheat detection context. Based on the aforementioned judgement and limitations in available time and computational resources, a full training routine of a DETR network from scratch will not be addressed in this study.

When looking at the network's structure, two main aspects have been targeted in order to provide a proper calibration of its behavior and generate a reasonable performance. Firstly,

the relevant hyperparameters had to be extracted from the model's full list of configurations. Since the focus was on a pre-trained model, any hyperparameter controlling the underlying network architecture has been excluded from the optimization schema. This schema only consisted of values describing the training strategy. Secondly, the image transformations used as a pre-processing step have been closely analyzed. Since the Wheat Head Dataset is known for having inconsistent images and bounding boxes [3], the suite of employed transformations has been reported to have a great influence on the behavior of a network used in this competition [1], regardless of the underlying model being YOLO, Faster-RCNN or DETR.

3.2 Prior frequency information

The strategy that this study will follow is based on adding prior frequency information about the target domain into the DETR network. More precisely, a Fast Fourier Transform (FFT) mask will be applied on the training images in order to extract additional domain knowledge and use it for further prediction. This filter is created by firstly applying FFT on the bounding box regions present in each image and averaging this decomposition over the entire dataset. Next, FFT is also applied on the entire image, again being averaged over the entire image set. The difference between these two frequency heat maps will be used for the final filter construction, while a specific threshold parameter will control how strong the difference needs to be between those two representations in order to have a specific frequency discarded or not. Lastly, a smoothing low-pass filter meant to curate high frequency noise left in the subtracted decomposition will be applied. Presented in Figure 2 is the conceptual process of mask creation.

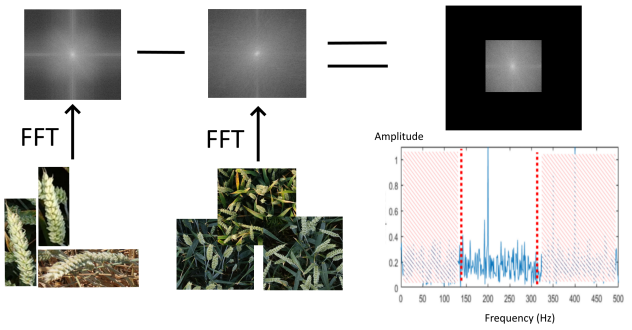

Figure 2: Creation of the Fourier mask. The interesting frequencies are the ones associated with the wheat heads inside the bounding boxes. The Fourier decomposition characterizing those bounding boxes is compared to the one applied on the entire image set. By subtracting the whole image Fourier average from the bounding box one, the particular bounding box frequencies are revealed. Lastly, frequencies lower than a certain value and higher than another are excluded, effectively applying a band-pass filter on the FFT difference representation.

[3]Some images in the training set have no annotated bounding boxes, while others have overlapping ones with extreme sizes - too small or too big for the contained element

Ultimately, two approaches have been explored for delivering the result of this additional FFT-mask filtering stage to the network. One of them is based on replacing the previous image with its filtered version, maintaining the RGB 3 channel DETR input type. The other one concatenates the original image with its filtered version and adds an extra convolutional layer for projecting from the resulting 6 channel input to the desired 3 channel one.

3.3 Evaluation process

The performance analyses executed for the two detection systems (original DETR vs FFT-based DETR) include a suite of various metrics [4] [5]:

- Kaggle score: $S = \frac{1}{|T|} \sum_{t \in T} \frac{TP(t)}{TP(t)+FP(t)+FN(t)}$
- Average precision: $AP = \frac{1}{|T|} \sum_{t \in T} \frac{TP(t)}{TP(t)+FP(t)}$
- Average recall: $AR = \frac{1}{|T|} \sum_{t \in T} \frac{TP(t)}{TP(t)+FN(t)}$

Training time (frames per second) and respective DETR losses have also been recorded. Additionally, a schema for visualizing encoder's self attention weights and encoder-decoder multi-headed attention fields for predicted objects is provided. It allows for capturing a part of the network's internal behavior, leading to a way of reasoning about the algorithm's ability for inference. Ultimately, for studying the dependent variable between the two DETR models, the previous performance metrics will be put into perspective in order to test the hypothesis. As to grasp a sense of their detection capability, both models will be briefly compared to YOLO-V3 and Faster R-CNN on the same wheat data set.

4 Experimental Setup and Results

In order to provide a valid experimental setup, the network had to be properly configured as to allow a relevant investigation of any changes resulted from adding FFT filtering. The overall calibration process targeted the image transformations pre-processing routine, the non-structural hyperparameter fixation, the Fourier mask creation parameters and the convolutional kernel for the channel projection layer.

For preparing the training environment, the given dataset has been first split into a test-train partition of 90/10. Parameter search has been performed with a stratified 5-fold cross-validation approach on the training set. The k-fold approach is well suited for the current context, since the dataset is quite scarce, with only 3.4k images. The stratified version ensures the same percentage of wheat head samples in both splits as in the complete set, accounting for the high variance in the numbers of targets across different images (from none to 72 maximum observed).

4.1 DETR

The image pre-processing stage starts with the visual transformations that have been used in the original DETR approach:

[4]T is the set of different IOU thresholds = [0.5-0.75— step = 0.005]

[5]TP - # true positives; FP - # false positives; FN - # false negatives

flip, resize, crop and normalization. Those original transformations were self-implemented, with only a few imported from corresponding standard libraries [6]. A crucial observation here is that the network could not run properly with the original implementation of the image transformations. This is due to the quality difference between the two datasets: COCO has a strict and well prepared training dataset, whereas Wheat Head Dataset also contains training images with no associated bounding boxes and even some bounding boxes with degenerate annotations. The manual functions used for initial image manipulation were trapped in undefined computations (nan and inf). Therefore, those low-level transformations have been substituted with a library-based equivalent of them, having the initial manual functions reproduced in a less verbose fashion [7].

Two models with different configurations were used for exploring DETR's performance on the Wheat Head Dataset. One of them is a baseline pre-trained adaptation of the original DETR code, called base-DETR. In order to grasp a sense of the native network's performance and set a starting point for its further optimization, minimal changes were brought to the model's logic. These involve the necessary modifications associated with the conversion from one dataset to another, including the number of target classes and the image adjustments [8]. Furthermore, the only hyper-parameter optimization performed for this model was for the learning schedule, which was applied with a batch step criterion, considering the expected small number of epochs required for convergence due to the pre-trained bonus. Fixing the training schedule increased the performance of this model by 11 S, due to moving the learning saturation point at a later epoch.

Parameter	base-DETR	ext-DETR
LR	1e-4	5e-5
LR_bb	1e-5	3e-5
Lr_drop	350	500
Lr_gamma	0.7	0.7

Table 1: Table presenting the hyper-parameter configurations for the baseline DETR (base-DETR) and for DETR with extended search on parameters (ext-DETR). All other parameters were kept the same as in the original model.
LR - learning rate for the transformer; LR_bb - learning rate for the CNN backbone; LR_drop - learning rate drop counter; LR_gamma - learning rate gamma for the decrease ratio

The second model, called ext-DETR, has been generated such as to make no presumptions on the proper calibration of the entire learning process. Therefore, the learning rates for the model's main components, the transformer and the CNN backbone, have also been included in the grid search optimization schema, along with the previously mentioned learning schedule. Finally, the two models' setup is presented in Table 1 and their results in Figure 3 and Table 2. The complete network configuration is available in the associated code repository.

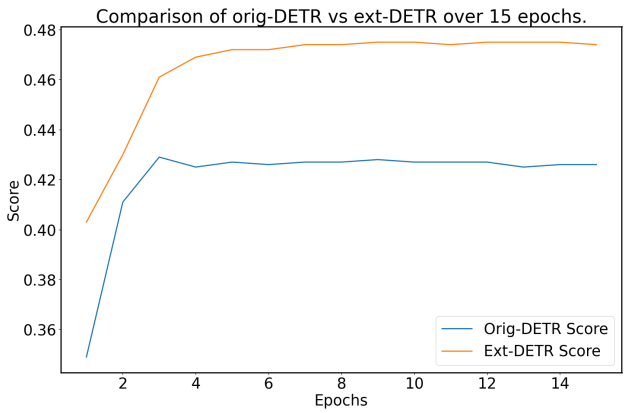

Figure 3: Performance comparison between base-DETR and ext-DETR over 15 epochs. It can be observed that the learning saturates around epoch 10 and therefore the final versions are stopped before this point for both models. Ext-DETR manages to achieve 4 S improvement over the baseline variant. Also, base-DETR initially grows and then flattens out faster than the extended version, correlating its higher learning rate with overshooting the global optimum and getting stuck in a local one.

Model	S	AP	AR
base-DETR	0.433	0.565	0.588
ext-DETR	**0.477**	0.606	0.634

Table 2: Results achieved by the two models (base-DETR and ext-DETR) on the test set. Ext-DETR manage to outperform base-DETR by 4 S points while having the overall AP/AR ratio kept the same.

The CNN backbone used was the ResNet50 non-dilated network for both models. The associated costs with the loss functions were kept unchanged. One significant aspect that was influencing the network's behavior for ext-DETR, making it generate too many bounding box class predictions, was the number of queries. Changing it from the original meant changing the dimensions of the positional embeddings and losing all their pre-learned weights. This was causing significant performance degradation of more than 15 AP and relatively low AR variations, showing an increased number of FPs correlated with the detector that was generating way more predictions than it was required.

4.2 FFT + DETR

The main focus of this research relied on enhancing the DETR network with an initial image processing step where the FFT-based filtered version of the training images is generated. From the 2 models previously introduced, the better performing one (ext-DETR) has been chosen for studying further in depth the effect of FFT prior knowledge. In this respect, the setup consisted of analyzing the influence that the

[6] The initial visual augmentation routine was based on PIL[8] image format and Torchvision[13] processing functions.

[7] New visual augmentation stage relied on OpenCV[3] image format and Albumentations[2] functions

[8] Provided images have been adjusted to have the same properties as the ones used for the initial network training: scale, color channel, annotations

parameter controlling the Fourier mask creation has over the image and ultimately over the network.

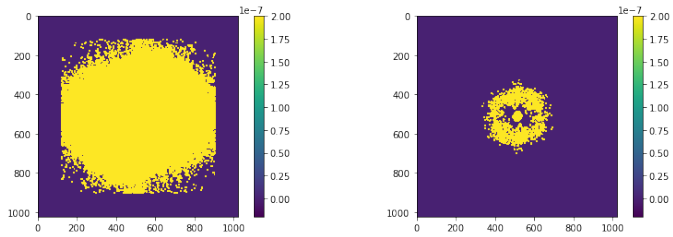

Figure 4: Effect of the threshold value on the amount of frequencies that are kept from the wheat head Fourier decomposition. The higher the threshold value, the stricter the mask becomes when applied on a Fourier domain representation, discarding more constituting frequencies from the image that is filtered.
Executed for the red channel on the 1e-8 threshold scale. Left: threshold = 2e-8; Right: threshold = 8e-8

The frequency-based filter is constructed from the difference between predominant wheat frequencies and the ones characterizing the whole image. The threshold value, which controls the required strength of this difference in order to mask a certain frequency, influences the amount of information that is being removed from the entire image. The effect of this threshold on the shape of the mask can be observed in Figure 4.

Model	configuration
FFT-DETR + Proj-FF-DETR	threshold = 2e-12 exclude = 120
Proj-FFT-DETR	kernel size = (9, 9); stride = 1

Table 3: Additional configuration of the FFT-based models on top of the ext-DETR one. Both FFT versions use the same mask and Proj-FFT-DETR has an extra initial convolutional layer with a square kernel of 9, which has been selected from the list of tested kernels with shapes varying from 1 to 15.

For injecting the details extracted by the mask further into the network, two approaches have been investigated: one in which the filtered image simply replaces the original input and another where the information from both is concatenated into 6 channels and further aggregated by a new convolutional layer into the 3 channel input that DETR is expecting. The two models are called FFT-DETR and Proj-FFT-DETR respectively. FFT-DETR limits the information that is provided to the model, aiming to remove large part of the background in order to concentrate the data around the wheat head particularities. On the other hand, Proj-FFT-DETR increases the amount of information that is given to the model by combining all the original image representation with the target object details on top of it. The concatenated 6 channel input is then projected by a convolutional kernel back into a standard 3 channel one, effectively having both image representations compressed into a single composite one. This way, the network receives the supplementary frequency features alongside with the rest of the original image, having an increased flexibility for discovering an optimal internal state from hidden patterns in the data. The proper configuration of both models is presented in Table 3.

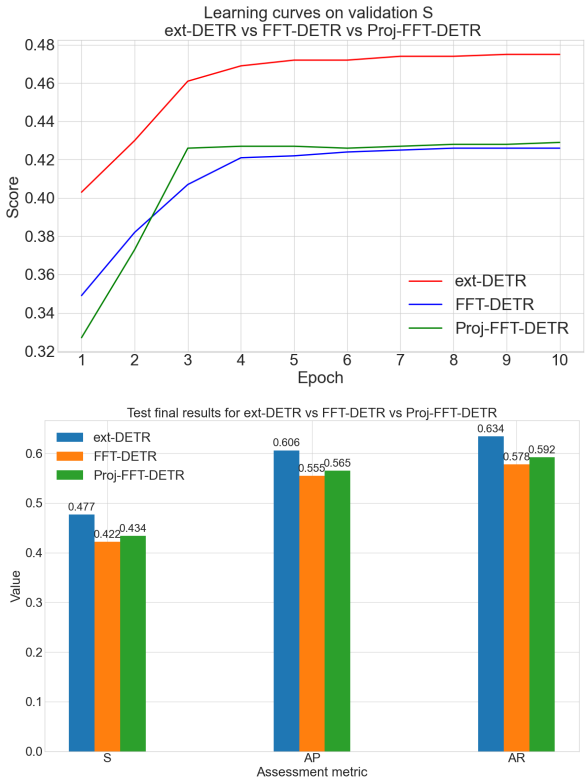

Figure 5: Comparison ext-DETR VS FFT-DETR vs Proj-FFT-DETR.
Top: Learning curves based on validation S for ext-DETR vs FFT-DETR vs Proj-FFT-DETR. Both FFT-DETR and Proj-FFT-DETR achieve similar results, while ext-DETR outperforms them by 4 S. Proj-FFT-DETR starts at the lowest initial value and has the greatest slope for the first 2 epochs, flattening out at the 3rd epoch and saturating around the 8th one.
Bottom: Results on final test set for S, AP, AR metrics. Largest gap between the models is on the AR metric, showing that ext-DETR reaches a better score due to a lower, more refined number of bounding box predictions.

The correlation between the mask creation parameters and the model's performance has been explored based on the simpler FFT-DETR over the stratified 5-fold cross-validation schema. An interesting observation emerged regarding the suitable threshold value of 2e-12. On further analysis of the suppressed frequency space, this optimal value lies somewhere at the middle. This means that the mask generated still allowed half of the auxiliary information to pass through the filter, which ultimately indicated that some of the background frequencies were important for the model's inference capability, contributing to its final performance. Additionally, the second parameter that describes a low-pass noise reduction filter has been fixed at 120, without any significant impact on the consequent performance.

Both FFT-based models along with ext-DETR have been exposed to a performance comparison, which is highlighted

Model	S	AP	AR
ext-DETR	**0.477**	**0.606**	0.634
FFT-DETR	0.422	0.55	0.57
Proj-FFT-DETR	0.434	0.565	0.592

Table 4: Final results achieved by the three models (ext-DETR, FFT-DETR and Proj-FFT-DETR) on the test set. The top performer is ext-DETR, with Proj-FFT-DETR following by a margin of 4 S and lastly FFT-DETR with a 1 S difference from the Proj-FFT-DETR. None of the models managed to break the 50 AP barrier and only ext-DETR reaches over 60 AP.

in Figure 5 across 10 epochs. The final results achieved by those predictors on the test set are underlined in Table 4, thus setting the benchmark for every model.

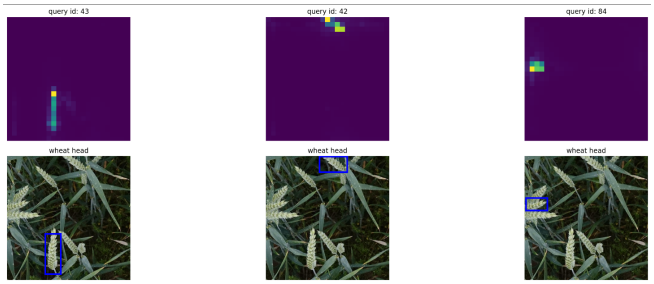

Figure 6: Visualization of encoder-decoder multi-head attention weights for a set of 3 predicted objects on a validation image. The model used is Proj-FFT-DETR with restoration of the initial unfiltered image after each prediction. The model learns to attend differently at the wheat head extremities then the way it does at the body of the spike.

For investigating the learning process based on the internal representation of the domain objects, attention maps have been generated for the encoder in Figure 7 and for the decoder in Figure 6, both belonging to the Proj-FFT-DETR model. The transformer behaviour visualization reveals that the network captures the most relevant regions for wheat heads that are in less crowded areas, while some background patches are misinterpreted by the encoder's attention layer as clusters of target objects.

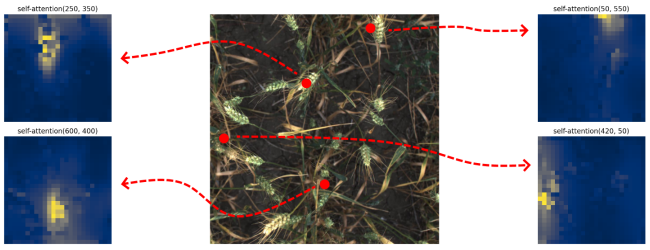

Figure 7: Visualization of encoder self-attention weights for a set of reference points inside the image. The encoder is capable of identifying some separate instances of wheat heads, while also struggling with a more diffused view when multiple objects are gathered together. Representation extracted during the execution of Proj-FFT-DETR on a validation image.

5 Discussion

The performance of the best FFT-based model, Proj-FFT-DETR, still remains lower than the one of ext-DETR by a small margin. Adding the mask filtering process slightly hinders the inference process by reducing the performance with 4 S and 5 AP. Combined with the 1.6X increase in training time, due to application of the mask for each input image, this approach proves to not be well suited for improving the prediction of the DETR network.

5.1 Model behaviour

A possible reason for the recorded behavior could lie in the underlying capabilities of the network's CNN backbone. Since its responsibility is to extract different image features by applying stacked convolutional kernels, it is possible that in a segment of its multi-function layers the network already learns to perform a procedure similar to FFT. This would mean that our intention of providing additional domain context is actually hurting the CNN's potential by reducing the amount of data delivered to just the frequency filtered one, effectively removing most of the background information.

The moment when the FFT mask is applied has been observed to have an effect on the results produced by the network. More specifically, applying the filter after the image transformations leads to a prediction quality decrease of 6 S and 5 AP than doing it before those. This could be explained by the non-commutativity of function composition between Fourier transform and random image crop. Since the mask has dimension (1024, 1024), cropping the image results in a necessary padding in order to bring it to the same shape. The padded regions, although carrying no actual information, are treated as white pixels (0, 0, 0) and therefore interfere with the Fourier decomposition, leading to a different result than the one obtained had the mask been applied before the crop. Since Proj-FFT-DETR needs post-transform frequency filtering, in order to keep both the transformed original image and the filtered version of it, the aforementioned observation could justify part of its performance.

The potential of DETR has not yet reached the state in which the baseline model can be used for accurate object detection tasks in domains different than the one COCO is based upon. The result of this study in the agricultural field, while using the pretrained model with transfer learning, shows a maximum of 60 AP on wheat head detection, with the FFT enhancements proving to reduce the performance in the end by a small amount. This result is relatively in line with the one obtained by DETR on crowd pedestrian detection [10], where the baseline model is applied on a similarly challenging dataset and achieves a maximum of 66 AP. Here, the structural limitation of the current model is pinpointed as the slow convergence of the attention modules, due to rectified attention units. On top of this, [18] reinforces this idea and highlights the redundancy in the transformer double layer. As a result, a possibility is that regardless of the FFT approach used, without addressing the structural limitation in any way, the model would still struggle to achieve a better result.

5.2 Image augmentations

A suite of image augmentations successfully used on the Wheat Head Dataset is provided by numerous Kaggle notebooks that participated in the Wheat Head competition. Across different implementations and variations, some transformations tend to be part of every accomplished detector with score above 60 S. Those pre-processing image transformations are more complex and diverse than the ones present in the original DETR code. Among them, special color filters (Gaussian noise, Random blur and RGB shift) have a particular significance for improving the detection capability of a model, due to the fact that images are taken at various lightning conditions. Additionally, cutouts are important for compensating the occasional extreme dimensions of the bounding boxes.

Conversely, the image color processing functions would not be compatible with the FFT paradigm. Since the Fourier decomposition captures the variance of pixel values, changing these pixels would result in a divergent frequency map that would not be able to capture the target object information as accurately as before.

6 Responsible Research

From an ethical standpoint, no immediate severe issues seem obvious while developing the current research study. The importance of responsible Artificial Intelligence (AI) is reflected in the current work, by placing the goal of this study not only on the scientific curiosity fueled by it, but also on its actual day-to-day application of a capable object detector in the field of precision agriculture.

With respect to the norms of valid academic research, two important aspects pertaining to reproducibility are to be mentioned. First, since the topic of this research lies in the domain of AI, the output values recorded are generated by a stochastic process and cannot be precisely simulated and reproduced. During one run, a neural network might get stuck in a local optimum and miss the rest of the parameter hyperspace, ending with values that are different than those from the previous identical run. The unpredictability of those results needs to be acknowledged and addressed by focusing on reporting outputs averaged over multiple runs instead of singleton training schedules. Therefore, aside from longer training sessions, the rest of the reported values have been produced by averaging over multiple cross-validation folds. Additionally, the existing randomness inside the algorithm is controlled by having a specific seed for the generator, leaving the uncertainty only on the inherent non-deterministic Machine Learning approach.

Second, most of the presented results rely on parts of code extracted from various open sources. The Kaggle community and the submissions made for Wheat Head Competition have been of tremendous help for putting in place the proper configuration of the dataset and the logistics behind different parts of training a Machine Learning model (k-fold validations, train/valid splits, inference functions and performance estimators). Additionally, the original DETR code has represented the main foundation on which various extra contributions have been brought by this study. Also, the code on which this research process has built contains all the necessary annotations and highlights so that the flow of logic is understandable and proper credits are given to different sources.

Moreover, in order to provide a complete overview over the environments in which the tested model has been run, the entire configuration schema is made available in the repository associated with this study. All the parameter optimization processes have been thoroughly documented and full training routines have been recorded with intermediate outputs. Regarding the associated dataset, a train-test split of 90/10 has been performed from the very beginning so that all the training/optimization has always been executed on the training set.

7 Conclusions and Future Work

The novel DETR object detector has been introduced for the task of precision agriculture, which was placed in the context of a Kaggle competition for wheat head detection. This end-to-end, anchor-free framework uses attention mechanisms for focusing the internal inference operation on different regions of the input, effectively managing to correlate neighbouring pixels.

With a minimum extra configuration from the already provided pre-trained version, this model manages to achieve 0.47 S & 0.60 AP [9]. Further incorporation of an FFT-based mask oriented towards filtering out background information is explored with two variations. One model has the initial images replaced by the FFT-filtered version of them. Another has the filtered information concatenated with the original one, followed by a convolutional layer projection of the resulting 6 channels back into the 3 channel input accepted by DETR. The results obtained indicate that injecting prior frequency knowledge into DETR does not improve its performance on wheat head detection. Both FFT-based models perform slightly worse than the plain DETR. Adding frequency information in the form of targeting the wheat heads with the FFT mask reduces the performance of the model by 4 S & 5 AP, while increasing the training schedule by a factor of 1.6x. The best results achieved by the Fourier DETR are 0.434 S & 0.565 AP.

Nevertheless, the full capabilities of those models have not been entirely explored, leaving some space for further hyper-parameter optimization and for a more suitable configuration of a longer learning strategy. Putting the results into perspective, the best models in the Kaggle competition, adaptations of Yolo-V3 and Faster R-CNN, accomplish a highest score of 0.74.

Two areas of further exploration present themselves as encouraging. The DETR framework could be run with an entirely custom learning schedule, with no pre-trained weights and all hyper-parameters carefully picked. This would imply a compute-intensive task, which extrapolated from the DETR's original training, could take up to 4 days for running through 400 epochs. Lastly, the image augmentations have proven to carry a significant influence on the behavior of the network. They need to be reasonably applied on the FFT-masked images in order to enhance the network's inference ability.

[9] see Subsection 3.3 for detailed metrics explanation.

References

[1] Global wheat detection challenge, Aug 2020. Published in Kaggle https://www.kaggle.com/c/global-wheat-detection.

[2] E. Khvedchenya V. I. Iglovikov A. Buslaev, A. Parinov and A. A. Kalinin. Albumentations: fast and flexible image augmentations. *ArXiv e-prints*, 2018.

[3] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.

[4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers, 2020.

[5] Jose Carranza-Rojas, Saul Calderon-Ramirez, Adán Mora-Fallas, Michael Granados-Menani, and Jordina Torrents-Barrena. Unsharp masking layer: Injecting prior knowledge in convolutional networks for image classification. In *Lecture Notes in Computer Science*, pages 3–16. Springer International Publishing, 2019.

[6] Wenlin Chen, James Wilson, Stephen Tyree, Kilian Q. Weinberger, and Yixin Chen. Compressing convolutional neural networks in the frequency domain. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, August 2016.

[7] Kamran Chitsaz, Mohsen Hajabdollahi, Pejman Khadivi, Shadrokh Samavi, Nader Karimi, and Shahram Shirani. Use of frequency domain for complexity reduction of convolutional neural networks. In *Pattern Recognition. ICPR International Workshops and Challenges*, pages 64–74. Springer International Publishing, 2021.

[8] Alex Clark. Pillow (pil fork) documentation, 2015.

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015.

[10] Matthieu Lin, Chuming Li, Xingyuan Bu, Ming Sun, Chen Lin, Junjie Yan, Wanli Ouyang, and Zhidong Deng. Detr for crowd pedestrian detection, 2021.

[11] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. *Computer Vision – ECCV 2014*, page 740–755, 2014.

[12] Varsha Nair, Moitrayee Chatterjee, Neda Tavakoli, Akbar Siami Namin, and Craig Snoeyink. Optimizing CNN using fast fourier transformation for object recognition. In *2020 19th IEEE International Conference on Machine Learning and Applications (ICMLA)*. IEEE, December 2020.

[13] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.

[14] Muhammad Taufiq Pratama, Sangwook Kim, Seiichi Ozawa, Takenao Ohkawa, Yuya Chona, Hiroyuki Tsuji, and Noriyuki Murakami. Deep learning-based object detection for crop monitoring in soybean fields. *2020 International Joint Conference on Neural Networks (IJCNN)*, 2020.

[15] Sanaz Rasti, Chris J. Bleakley, Guénolé C. M. Silvestre, N. M. Holden, David Langton, and Gregory M. P. O'Hare. Crop growth stage estimation prior to canopy closure using deep learning algorithms. *Neural Computing and Applications*, 33(5):1733–1743, June 2020.

[16] M. Shahbandeh. Wheat - statistics and facts, Mar 2021. Published in Statista https://www.statista.com/topics/1668/wheat/.

[17] A. Srinivasan, A. Srikanth, H. Indrajit, and V. Narasimhan. A novel approach for road accident detection using detr algorithm. In *2020 International Conference on Intelligent Data Science Technologies and Applications, IDSTA 2020*, pages 75–80, 2020.

[18] Zhiqing Sun, Shengcao Cao, Yiming Yang, and Kris Kitani. Rethinking transformer-based set prediction for object detection, 2020.

[19] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2017.