Vol. 1, Issue 11 | Nov 2026 | Open Access — Research Paper

AI-DRIVEN PALM OIL FRUIT QUALITY ASSESSMENT USING COMPUTER VISION AND VISION TRANSFORMERS (ViTs)

B Ravindra Naik, Manne Ganesh, Nunna Sai Naresh, Nelaveni Vivek Vardhan
Department of CSE (AI & ML), CMR Technical Campus, Hyderabad, Telangana, India – 501401

***Abstract*—** Harvesting an oil palm fresh fruit bunch (FFB) at the right stage of ripeness decides how much oil it yields and how many free fatty acids the oil contains, yet in most plantations ripeness is still judged by eye. Ripeness stages are ordered: an immature bunch differs more from a decayed one than from a partially ripe one, but standard image classifiers treat the stages as unrelated labels. We describe an AI-driven fruit quality assessment system that grades ripeness from a photograph using computer vision. It replaces the classification head of a Vision Transformer (ViT-B/16) with an ordinal regression head, so that the model predicts a single continuous ripeness score from 0 (immature) to 4 (decayed). The score is rounded to one of five stages and mapped to harvest advice, and it is served through a web application that grades an uploaded photograph. We evaluate on a public dataset of 4,728 field images with predefined training, validation and test splits. While completing the project we found that the original training script ignored these splits, so most test images had been used for training, and that its data augmentation had been switched off; both faults are corrected. On a laptop without a GPU we report the frozen-backbone configuration of the model: over five random seeds the ordinal head reaches 82.4 ± 1.0% exact accuracy, 99.4 ± 0.2% accuracy within one stage, a mean absolute error of 0.182 stages and a quadratic weighted kappa of 0.924 on the 718 held-out test images, compared with 90.8 ± 0.6% exact accuracy and a kappa of 0.935 for a five-way softmax head trained on the same features. The softmax head is thus the more accurate grader, while the ordinal head makes about four times fewer errors that span two or more stages (0.6% against 2.5% of test images); its continuous score preserves the order of the stages but compresses the upper end of the scale.

***Index Terms*—**Oil Palm, Fruit Quality Assessment, Computer Vision, Vision Transformer, Ripeness Grading, Ordinal Regression, Precision Agriculture

I. INTRODUCTION

Palm oil is the most widely produced vegetable oil in the world, and Indonesia and Malaysia together supply most of it. The quality of the oil is largely decided in the field, when a harvester chooses which fresh fruit bunches (FFBs) to cut. A bunch cut too early has not finished accumulating oil; a bunch cut too late loosens its fruitlets, which fall and are lost, and the bruised fruit raises the free fatty acid content of the oil. Mills therefore grade incoming bunches and penalise unripe and overripe deliveries.

In practice the grading is visual. A harvester or grader looks at the colour of the fruitlets and the number of loose fruits and assigns a stage. The judgement is subjective, depends on lighting and experience, and is hard to apply consistently across thousands of bunches a day. Sensor-based methods such as hyperspectral imaging [2] and fluorescence measurement [1] are more objective, but they need specialised equipment. An ordinary photograph taken with a phone is far cheaper, which makes image-based grading attractive [3], [4].

Ripeness is not a set of unrelated categories. The five stages used here, immature, partially ripe, fully ripe, overripe and decayed, form a scale, and the cost of a mistake grows with its distance on that scale: calling a fully ripe bunch partially ripe delays a harvest by a few days, while calling it immature or decayed changes the decision entirely. A softmax classifier trained with cross-entropy ignores this structure, since every wrong label is penalised equally. Ordinal regression [5]–[7] keeps the order by predicting a position on the scale instead of a label.

We combine an ordinal regression head with a Vision Transformer (ViT) [8]. A ViT splits the image into patches and relates every patch to every other through self-attention [9], so it can weigh the colour of individual fruitlets against the appearance of the whole bunch, which is how a human grader works. The model outputs a single ripeness score, which also gives a natural basis for harvest advice. The contributions of this work are:

1) An ordinal ripeness model that replaces the ViT-B/16 classification head with a regression head trained with the Smooth L1 loss, and a rule that maps the continuous score to five stages and three harvest actions.

2) A correction of the project's original training pipeline, which ignored the dataset's predefined splits and disabled augmentation, together with an evaluation protocol in which the test split is used only once.

3) A comparison of the ordinal head with a five-way softmax head and a linear probe on the same frozen features, over five random seeds, on 718 held-out test images.

4) A web application that grades an uploaded photograph and shows the stage, the continuous score and the harvest advice.

Section II reviews related work, Section III describes the dataset and the evaluation protocol, Section IV presents the methodology and Section V the implementation. Section VI reports and discusses the results, and Section VII concludes.

II. RELATED WORK

A. Ripeness Grading of Oil Palm Bunches

Early work measured ripeness with dedicated sensors. Hazir et al. [1] related flavonoid and anthocyanin content,

measured with a hand-held fluorescence sensor, to ripeness and reported a best classification accuracy of 87.7%. Bensaeed et al. [2] graded fruit as underripe, ripe or overripe from hyperspectral reflectance with machine learning. These methods are objective but require instruments that few plantations have.

Camera-based grading has become practical with deep learning. Suharjito et al. [3] published annotated videos and images of FFB piles from mills in Indonesia, labelled in six categories, for training object detectors. Kamal et al. [4] released the dataset used in this work: images of single bunches photographed in the field, labelled on an explicit five-level ordinal scale and distributed with stratified training, validation and test splits. Most published models on such data are convolutional classifiers or detectors that treat the stages as independent classes.

B. Vision Transformers

Dosovitskiy et al. [8] showed that a standard Transformer encoder [9] applied to a sequence of 16×16 image patches matches or exceeds convolutional networks such as ResNet [10] on image classification when pretrained on large datasets such as ImageNet [11]. Because self-attention connects all patches from the first layer, a ViT can combine local texture with global layout, which suits an object such as a fruit bunch whose ripeness is judged from the colour of many small fruitlets across the whole bunch.

C. Ordinal Regression

Ordinal regression predicts labels that have a natural order. Frank and Hall [5] reduced it to a series of binary "greater than k" problems; Niu et al. [6] trained a convolutional network with one output per threshold for age estimation; and Cao et al. [7] made the threshold outputs rank-consistent. The simplest form, which we use, treats the stage index as a real number and regresses it directly. This gives one interpretable score, needs a single output unit, and penalises an error in proportion to its distance on the scale.

D. Positioning

Table I summarises how the proposed design differs from the approaches above. The comparison describes typical configurations of each approach rather than particular published systems.

TABLE I
DESIGN COMPARISON OF RIPENESS GRADING APPROACHES

Property	Manual grading	Sensor-based	CNN classifier	Proposed
Equipment	None	Spectrometer / sensor	Camera	Camera
Objective and repeatable	No	Yes	Yes	Yes
Uses the order of stages	Implicitly	Rarely	No	Yes
Global context of the bunch	Yes	No	Limited	Yes (self-attention)
Single score for advice	No	No	No	Yes

III. DATASET

A. Description

We use the ordinal oil palm ripeness dataset of Kamal et al. [4]. It contains 4,728 RGB images of single FFBs photographed in the field in Central Kalimantan, Indonesia, under natural light and with cluttered backgrounds of soil, leaves and other bunches. Every image is 256×256 pixels and is labelled with one of five ordered stages, stored in folders whose names carry the rank (0Immature to 4Decayed). Fig. 1 shows examples.



Fig. 1. Two example images of each ripeness stage from the dataset.

The dataset is distributed with three text files that list the images of the training, validation and test splits, stratified by class in the proportion 70:15:15. Table II and Fig. 2 give the counts. The classes are imbalanced: fully ripe and partially ripe bunches make up 61% of the images, and decayed bunches only 8.7%.

TABLE II
IMAGES PER STAGE AND SPLIT

Stage (rank)	Training	Validation	Testing	Total
Immature (0)	530	113	115	758
Partially Ripe (1)	982	210	213	1,405
Fully Ripe (2)	1,042	222	225	1,489
Overripe (3)	465	98	102	665
Decayed (4)	287	61	63	411
Total	3,306	704	718	4,728

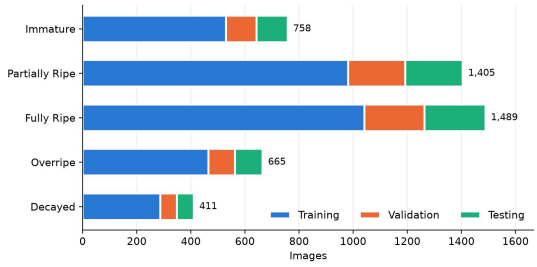


Fig. 2. Number of images per stage, divided into the three predefined splits.

B. Data Leakage in the Original Pipeline

The first version of the training script did not read the split files. It loaded all 4,728 images and divided them 80:20 with an unseeded random split, so the official test images were spread across its training and validation sets. With 80% of images going to training, about 574 of the 718 official test images are expected to have been trained on, and because the split was not seeded, which ones cannot be recovered. A test score for that checkpoint would therefore mostly measure memorisation.

The same script also assigned the validation transforms to the dataset object shared by both subsets of the split, which silently replaced the training transforms. The model was therefore trained without the flips, rotations and colour changes that the code appeared to apply. We corrected both faults: the training, validation and test sets are now built from the three split files, each with its own transforms, and the random seed is fixed.

C. Evaluation Protocol

Models are trained on the training split, all choices that depend on data (early stopping and the selected epoch) are made on the validation split, and the test split is scored once at the end. Besides exact accuracy we report the share of predictions within one stage of the truth (off-by-one accuracy), the mean absolute error (MAE) in stages, the macro-averaged F1 score, which weights the small decayed class equally with the others, and the quadratic weighted kappa (QWK) [12], which measures agreement on an ordinal scale and penalises an error by the square of its distance:

$$QWK = 1 - \Sigma\, w_{ij}\, O_{ij} / \Sigma\, w_{ij}\, E_{ij},\ w_{ij} = (i-j)^2 / (K-1)^2 \qquad (1)$$

where O is the observed confusion matrix, E the matrix expected by chance from the marginal totals, and K = 5.

IV. PROPOSED METHODOLOGY

A. System Architecture

Fig. 3 shows the pipeline. An image is resized and normalised, passed through the ViT-B/16 backbone, and the 768-dimensional [CLS] embedding is mapped by the ordinal head to a single ripeness score s. A fixed rule converts s into a stage and a harvest action.

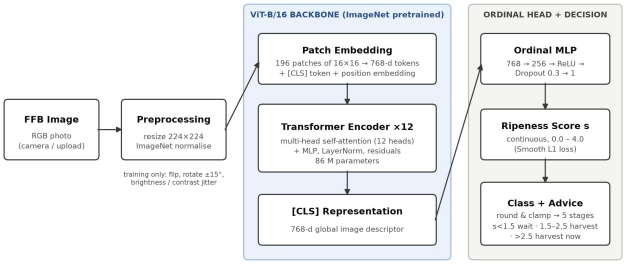


Fig. 3. Architecture of the proposed system. The ViT-B/16 backbone produces a global image descriptor; the ordinal head maps it to one ripeness score, which is converted into a stage and harvest advice.

B. Preprocessing and Augmentation

Images are converted to RGB, resized to 224×224 pixels and normalised with the ImageNet channel means and standard deviations, as the pretrained backbone expects. During training only, each image is flipped horizontally with probability 0.5, rotated by up to 15° and given random brightness and contrast changes of up to 20%, to simulate the different camera angles and lighting found in a plantation. Validation, test and application images receive only the deterministic resize and normalisation.

C. Vision Transformer Backbone

We use the ViT-B/16 model of torchvision [13], pretrained on ImageNet-1k [11]. The 224×224 input is cut into 196 non-overlapping patches of 16×16 pixels; each patch is projected to a 768-dimensional token, a learnable [CLS] token is prepended and position embeddings are added. Twelve Transformer encoder layers, each with 12-head self-attention, an MLP block, layer normalisation and residual connections, process the 197 tokens. The final [CLS] token summarises the whole image. The backbone has 85.8 million parameters.

D. Ordinal Regression Head

The original classification layer is replaced by a small multilayer perceptron: a linear layer from 768 to 256 units, a ReLU, dropout with probability 0.3 and a linear layer to one output. The head adds 196,865 parameters. For an image x with stage $y \in \{0, 1, 2, 3, 4\}$, the network predicts a real-valued score

$$s = f(x),\ d = s - y \qquad (2)$$

and is trained with the Smooth L1 (Huber) loss [14], which is quadratic for small errors and linear for large ones, so that a few mislabelled or ambiguous images do not dominate the gradient:

$$\ell(d) = 0.5\, d^2 \text{ if } |d| < 1,\ |d| - 0.5 \text{ otherwise} \qquad (3)$$

Because the target is the stage index itself, predicting stage 4 for an immature bunch costs four times as much as predicting stage 1, unlike cross-entropy, which charges both mistakes the same.

E. Decision Rule and Harvest Advice

At inference the score is rounded to the nearest integer and clamped to the valid range, $\hat{y} = \min(4, \max(0, \text{round}(s)))$, giving the stage. The continuous score is also kept, because it shows how far a bunch is from the next stage. Table III lists the harvest advice derived from it.

TABLE III
RIPENESS STAGES AND HARVEST ADVICE

Score s	Stages	Advice
$s < 1.5$	Immature, Partially ripe	Do not harvest; the fruit needs more time
$1.5 \le s \le 2.5$	Fully ripe	Prime condition for harvest
$s > 2.5$	Overripe, Decayed	Harvest immediately; the fruit is degrading

F. Training Configurations

We define two configurations of the same model (Table IV). In full fine-tuning all weights are updated with AdamW [15] for 15 epochs, and the checkpoint with the lowest validation MAE is kept. This needs a GPU: on the laptop used for this work, an Intel Core i3-1305U without a GPU, one optimisation step on 16 images took about 105 s, or roughly six hours per epoch. In the frozen-backbone configuration the backbone is not trained. Each image is passed through it once, the [CLS] embeddings are cached, and only the head is trained on them, which takes seconds. All results in Section VI use this configuration; full fine-tuning is implemented and left for a GPU run.

TABLE IV
TRAINING CONFIGURATIONS

Setting	Full fine-tuning	Frozen backbone
Trainable parameters	86.0 M	0.20 M (head)
Optimiser	AdamW, weight decay 0.01	AdamW, weight decay 0.01
Learning rate	1×10^{-4}	1×10^{-3}
Batch size	16	64
Epochs	15	up to 150, early stopping (patience 25)
Augmentation	Flip, rotation, colour jitter	None (features cached once)
Model selection	Lowest validation MAE	Lowest validation MAE
Hardware	GPU recommended	Laptop CPU

V. IMPLEMENTATION

A. Software Modules

The system is written in Python with PyTorch [16] and torchvision; metrics use scikit-learn [17]. Table V lists the modules.

TABLE V
PROJECT MODULES

Module	Responsibility
dataset.py	Loads images and ordinal labels, optionally restricted to a split file
model.py	ViT-B/16 backbone with the ordinal head
train.py	Full fine-tuning on the official training and validation splits
extract_features.py	Caches frozen [CLS] embeddings for all splits
train_head.py	Trains the ordinal, softmax and logistic heads over five seeds
evaluate.py	Test metrics, confusion matrix and score plots
predict.py	Command-line prediction for one image
app.py	Streamlit web application

B. Web Application

The web application is built with Streamlit. It loads the model once and caches it between requests. A user uploads a JPEG or PNG photograph of a bunch; the application applies the same deterministic preprocessing as validation, runs the model and shows the image next to the predicted stage, the continuous score on a 0–4 progress bar and the harvest advice of Table III (Fig. 4). The trained head can be exported as a complete model checkpoint, so the application, the command-line tool and the evaluation script all load the same file format.

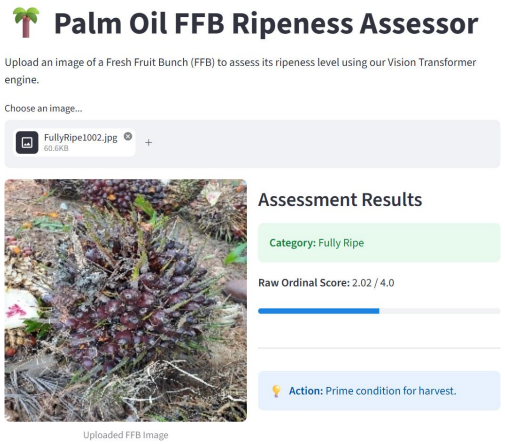


Fig. 4. Web application grading an uploaded photograph as fully ripe (score 2.02) with the advice to harvest.

C. Inference Cost

Extracting features for all 4,728 images took 117 minutes on the laptop CPU, about 1.5 s per image, while other applications were running. The head itself costs a fraction of a millisecond. On a GPU the backbone runs in milliseconds per image, so the CPU figure is an upper bound for a deployment.

VI. RESULTS AND DISCUSSION

A. Experimental Setup

All heads are trained on the 3,306 training images, selected on the 704 validation images and evaluated once on the 718 test images, using the frozen ViT-B/16 features. The ordinal and softmax heads have the same hidden layer and differ only in the output layer and the loss. Each is trained with five random seeds (0 to 4), and we report the mean and standard deviation over seeds. The linear probe is a multinomial logistic regression on standardised features. For the softmax head the MAE uses the arg-max class.

B. Comparison of Prediction Heads

TABLE VI
RESULTS ON THE 718 TEST IMAGES (MEAN ± STD OVER FIVE SEEDS)

Head	Exact acc. (%)	Within one (%)	MAE	Macro-F1	QWK
Ordinal regression (proposed)	82.4 ± 1.0	99.4 ± 0.2	0.182 ± 0.010	0.819 ± 0.005	0.924 ± 0.005
Softmax classifier	90.8 ± 0.6	97.5 ± 0.1	0.118 ± 0.007	0.910 ± 0.005	0.935 ± 0.003
Logistic regression (linear probe)	87.9	97.9	0.142	0.891	0.931

Table VI compares the three heads. The ordinal head reaches 82.4 ± 1.0% exact accuracy and places 99.4 ± 0.2% of test images within one stage of the truth. Compared with the softmax head trained on the same features, its exact accuracy is 8.4 percentage points lower, its MAE is 0.064 stages higher and its QWK is 0.011 lower. On frozen features the softmax head is therefore the more accurate of the two: it leads by about eight percentage points of exact accuracy and also has the lower MAE and the higher QWK, and even the

linear probe is more accurate than the ordinal head. The ordinal head is ahead on one measure only, the share of predictions within one stage (99.4% against 97.5%): when it is wrong, it is almost always wrong by a single stage. The differences are well outside the seed-to-seed variation, which is about one percentage point.

C. Errors by Stage

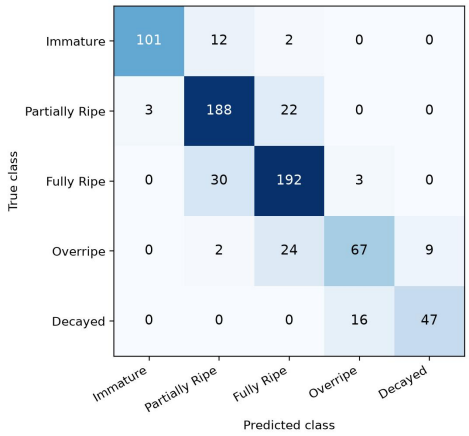


Fig. 5. Confusion matrix of the ordinal head (seed 0) on the 718 test images.

Fig. 5 shows the confusion matrix of the ordinal head for seed 0, which was fixed in advance rather than chosen by its result. Of its 123 errors, 119 (97%) are between neighbouring stages and 4 are two or more stages away. The errors concentrate where the stages are hardest to tell apart by eye: partially ripe against fully ripe, and overripe against its neighbours. Recall is lowest for overripe (0.657) and decayed (0.746) bunches. For comparison, the softmax head of the same seed makes 61 errors, but 17 of them are two or more stages away, among them eight overripe bunches predicted as partially ripe and three fully ripe bunches predicted as immature. In the field the second kind of error is the costly one, because it reverses the harvest decision.

TABLE VII
PER-STAGE RESULTS OF THE ORDINAL HEAD (SEED 0)

Stage	Precision	Recall	F1	Images
Immature	0.971	0.878	0.922	115
Partially Ripe	0.810	0.883	0.845	213
Fully Ripe	0.800	0.853	0.826	225
Overripe	0.779	0.657	0.713	102
Decayed	0.839	0.746	0.790	63

D. Distribution of the Ripeness Score

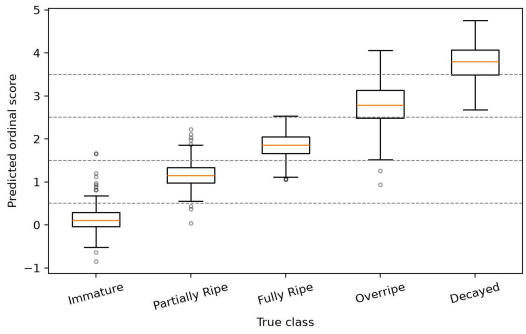


Fig. 6. Continuous ripeness score predicted for the test images of each stage. Dashed lines mark the rounding boundaries between stages.

Fig. 6 shows the continuous score by true stage. The medians rise steadily from about 0.1 for immature to about 3.8 for decayed bunches, so the score preserves the order of the stages and can be read as a degree of ripeness. The upper stages, however, are pulled towards the centre of the scale: the median is about 1.85 for fully ripe, 2.8 for overripe and 3.8 for decayed bunches, and roughly a quarter of the overripe and decayed images fall below their rounding boundary. Because the boundaries are fixed at 0.5, 1.5, 2.5 and 3.5, this downward shift turns directly into the low recall of the two last stages. A regression loss on an imbalanced scale, where the middle stages dominate, is known to produce such shrinkage.

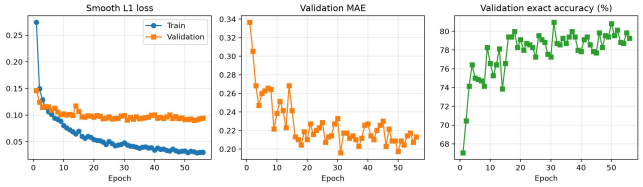


Fig. 7. Training of the ordinal head (seed 0): loss on the training and validation sets, and validation MAE and exact accuracy per epoch.

Fig. 7 shows the training of the ordinal head for seed 0. The training loss keeps falling, while the validation loss levels off after about 15 epochs; the lowest validation MAE (0.196) is reached at epoch 31, and early stopping ends training 25 epochs later. Validation exact accuracy settles at about 79–81%, close to the test result, so the selected head does not appear to overfit the validation set.

E. Discussion

Accuracy against severity of errors. The experiment does not support the expectation that an ordinal head is simply better. With a frozen backbone, five class-specific outputs separate the stages better than a single score with fixed, equally spaced boundaries. What the ordinal head buys is a different error profile: it almost never confuses stages that are far apart, and it gives a continuous score that shows how close a bunch is to the next stage. Whether that is worth eight points of exact accuracy depends on how the grade is used; for a harvest decision with three actions (Table III), the stage boundaries that matter are only those at 1.5 and 2.5.

Where the ordinal head loses. The score distribution suggests a concrete cause: the score is compressed at the top of the scale, and the fixed rounding boundaries do not compensate. Two standard remedies follow directly. The boundaries can be learned on the validation split instead of fixed at half-integers, and a rank-consistent formulation such as CORAL [7] replaces the single regression output with one output per boundary. We have not evaluated either, so we do not claim that they close the gap.

Frozen against fine-tuned backbone. All results here use ImageNet features that were never adapted to palm bunches. Fine-tuning the backbone, which the corrected training script supports, lets the features themselves become ordered by ripeness and may change the comparison between the heads. The numbers in Table VI should therefore be read as a lower bound for this architecture, measured honestly on unseen data, rather than as its best achievable accuracy.

F. Limitations

- All reported results use a frozen backbone. Full fine-tuning, which usually improves accuracy, is implemented but was not run, because the available laptop had no GPU.
- The checkpoint used by the web application in Fig. 4 was trained by the original script and is affected by the leakage described in Section III-B. Its predictions are suitable for demonstration only until it is retrained.
- The images come from one region of Indonesia and one data collection campaign. Performance on other cultivars, cameras and lighting has not been measured.
- The stage labels are single expert judgements on a visual scale; ambiguity between neighbouring stages is part of the data, and part of the remaining error may be label noise.
- The application grades one bunch per photograph and does not check whether the image shows a palm bunch at all.

VII. CONCLUSION AND FUTURE WORK

We built a ripeness assessment system for oil palm fresh fruit bunches in which a Vision Transformer produces a single ordinal ripeness score that is converted into a stage and a harvest action, and we served it through a web application. While completing the project we found that the original training script leaked most test images into training and trained without augmentation; after correcting both, every result is measured on images that the model has never seen.

On frozen ViT-B/16 features the ordinal head reaches 82.4% exact accuracy, places 99.4% of test images within one stage and achieves a quadratic weighted kappa of 0.924. A softmax head on the same features is more accurate (90.8%, kappa 0.935) but makes about four times as many errors that span two or more stages. The ordinal score is therefore the safer signal for a harvest decision but not the more accurate grade, and its weakness is a compression of the upper stages that learned boundaries or a rank-consistent head could address.

Future work will: (i) fine-tune the full ViT-B/16 on a GPU with the corrected training script and compare it with the frozen model; (ii) learn the stage boundaries on the validation split and test rank-consistent ordinal heads [7] and class-weighted losses for the small decayed class; (iii) evaluate on photographs from other plantations, devices and lighting; (iv) compress the model by distillation or quantisation so that it runs offline on a phone; and (v) combine the ordinal head with a detector to grade several bunches in one photograph.

REFERENCES

[1] M. H. M. Hazir, A. R. M. Shariff, and M. D. Amiruddin, "Determination of oil palm fresh fruit bunch ripeness—Based on flavonoids and anthocyanin content," *Industrial Crops and Products*, vol. 36, no. 1, pp. 466–475, 2012.

[2] O. M. Bensaeed, A. R. M. Shariff, A. B. Mahmud, H. Shafri, and M. Alfatni, "Oil palm fruit grading using a hyperspectral device and machine learning algorithm," *IOP Conference Series: Earth and Environmental Science*, vol. 20, 2014.

[3] Suharjito et al., "Annotated datasets of oil palm fruit bunch piles for ripeness grading using deep learning," *Scientific Data*, vol. 10, 2023, doi: 10.1038/s41597-023-01958-x.

[4] I. M. Kamal et al., "An ordinal dataset for ripeness level classification in oil palm fruit quality grading," Mendeley Data, V1, 2024, doi: 10.17632/424y96m6sw.1.

[5] E. Frank and M. Hall, "A simple approach to ordinal classification," in *Proc. 12th European Conference on Machine Learning (ECML)*, 2001, pp. 145–156.

[6] Z. Niu, M. Zhou, L. Wang, X. Gao, and G. Hua, "Ordinal regression with multiple output CNN for age estimation," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 4920–4928.

[7] W. Cao, V. Mirjalili, and S. Raschka, "Rank consistent ordinal regression for neural networks with application to age estimation," *Pattern Recognition Letters*, vol. 140, pp. 325–331, 2020.

[8] A. Dosovitskiy et al., "An image is worth 16x16 words: Transformers for image recognition at scale," in *Proc. International Conference on Learning Representations (ICLR)*, 2021.

[9] A. Vaswani et al., "Attention is all you need," in *Advances in Neural Information Processing Systems*, vol. 30, 2017, pp. 5998–6008.

[10] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[11] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009, pp. 248–255.

[12] J. Cohen, "Weighted kappa: Nominal scale agreement with provision for scaled disagreement or partial credit," *Psychological Bulletin*, vol. 70, no. 4, pp. 213–220, 1968.

[13] TorchVision maintainers and contributors, "TorchVision: PyTorch's computer vision library," 2016. [Online]. Available: https://github.com/pytorch/vision

[14] R. Girshick, "Fast R-CNN," in *Proc. IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 1440–1448.

[15] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *Proc. International Conference on Learning Representations (ICLR)*, 2019.

[16] A. Paszke et al., "PyTorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems*, vol. 32, 2019, pp. 8024–8035.

[17] F. Pedregosa et al., "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

© IJVAST — International Journal of Vast Advancements in Science and Technology.
ISSN: 2839-5379 | DOI: 10.68296/IJVAST20260092 | Open Access — CC BY 4.0 License