Multispectral Analysis of Plant Physiological Stress Using Deep Convolutional Neural Networks

Sathvik Koti
Department of Electronic Engineering,
Maynooth University, Ireland
Email: sathvikkir007@gmail.com

***Abstract*—This study presents a deep learning framework for the precise assessment of physiological stress in oats and barley, utilizing multispectral imaging and convolutional neural networks (CNNs). We processed multispectral video recordings—capturing RGB, near-infrared, and red-edge bands—to extract and align image frames with LI-600 Licor measurements of stomatal conductance and leaf temperature, collected from a controlled farmbed. A two-stream CNN, with an EfficientNetV2-B3 RGB encoder fused via FiLM gating with a CNN processing four spectral maps (NDVI, NDRE, z-scored NIR, z-scored red-edge) and an NDVI-based mask to suppress background, followed by a multilayer perceptron, was trained to predict these stress indicators using Huber loss (primary; MSE for leaf temperature in one variant). A threshold-based classifier derived drought and heat stress levels for practical interpretation. Sensor-assisted predictions using leaf vapor pressure deficit (VPD leaf) provided a diagnostic benchmark, demonstrating robust performance for leaf temperature but moderate results for stomatal conductance. Key contributions include automated video frame extraction and data alignment, advancing precision agriculture. This non-invasive approach supports sustainable crop monitoring, with plans to incorporate cost-effective ambient vapor pressure deficit (VPD ambient) measurements for future enhancements.**

***Index Terms*—Convolutional neural networks, multispectral imaging, plant physiological stress, precision agriculture.**

I. Introduction

The timely and accurate detection of plant stress is a critical challenge in precision agriculture, directly impacting crop yield and resource efficiency. Traditional methods, often manual or lab-based, are not scalable for large-scale operations due to their high cost and time consumption. This study addresses the research question: Can a two-stream convolutional neural network (CNN) that fuses RGB and spectral index maps (NDVI, NDRE, z-scored NIR, and z-scored red-edge) accurately predict stomatal conductance (gsw) and leaf temperature (Tleaf) in oats and barley grown in farmbeds? We hypothesize that multispectral fusion, using a two-stream CNN with Feature-wise Linear Modulation (FiLM) gating, yields higher accuracy than RGB-only or indices-only baselines and supports practical drought and heat stress binning via thresholds. By leveraging multispectral imaging and deep learning, this work advances non-invasive stress assessment in controlled farmbed environments.

A. Problem Statement and Motivation

1) Importance of Plant Stress Detection: Drought and heat stress reduce photosynthetic efficiency and water use in crops, leading to yield losses. For oats and barley, timely detection of stress indicators like gsw and Tleaf is essential for precision agriculture, enabling data-driven irrigation and environmental control in farmbeds.

Fig. 1. Farmbed Setup Used for Data Collection

2) Limitations of Traditional Methods: Conventional stress monitoring relies on manual measurements, such as those from the LI-600 Licor porometer/fluorometer, which are labor-intensive, time-consuming, and limited in scalability. These methods struggle to provide real-time, high-throughput assessments required for modern agriculture.

3) Role of Multispectral Imaging in Agriculture: Multispectral imaging, capturing RGB, near-infrared (NIR), and red-edge bands, offers a non-invasive solution. By deriving indices like NDVI and NDRE from video frames, it enables automated, scalable monitoring of plant health, aligning with the needs of precision agriculture in farmbed settings.

B. Literature Review

Recent advancements in deep learning (DL) have revolutionized plant stress detection, particularly through the integration of multispectral imaging. Comprehensive reviews highlight the evolution of DL techniques for this purpose. For instance, Altalak et al. [8] provide an overview of DL applications in plant stress imaging, emphasizing convolutional neural networks (CNNs) for feature extraction from hyperspectral and multispectral data. Similarly, Tanvir et al. [7] discuss state-of-the-art DL models for detecting biotic and abiotic stresses, noting challenges such as data scarcity and model interpretability. These reviews underscore gaps in multitask regression for physiological metrics like gsw and Tleaf, where most works focus on classification of visible symptoms rather than quantitative predictions.

Proximal sensing methods, including multispectral cameras, have been explored for real-time stress detection [9]. Studies like those by Costa et al. [10] demonstrate the use of UAV-based multispectral imaging for drought stress in crops, achieving high accuracy via vegetation indices but lacking fusion with DL for regression tasks. Our work fills this gap by proposing a two-stream CNN that fuses RGB and spectral indices via FiLM gating [1], improving over single-stream models in efficiency and accuracy for oats and barley. Compared to explainable DL pipelines for drought detection in potatoes [11], our approach incorporates sensor-assisted benchmarks using VPD leaf, providing a pathway to cost-effective VPD ambient integration. This addresses limitations in scalability and non-invasiveness identified in systematic reviews of multi-modal analytics [12], advancing precision agriculture by enabling continuous monitoring in farmbeds.

C. Original Contributions and Externally Provided Resources

1) Novel Contributions: This work introduces several contributions to plant stress analysis:

- An automated pipeline for extracting multispectral video frames and aligning them with LI-600 Licor measurements via timestamps.
- A two-stream CNN architecture combining an EfficientNetV2-B3 RGB encoder with a CNN processing four spectral index maps (NDVI, NDRE, z-scored NIR, z-scored red-edge), fused via FiLM gating for joint gsw and Tleaf regression.
- An NDVI-based foreground mask to suppress background noise, enhancing the model's focus on plant regions.
- A threshold-based classifier mapping regression outputs to practical drought and heat stress levels, supported by confusion matrices.

2) External Resources and Acknowledgments: This project utilized TensorFlow/Keras with EfficientNetV2 pre-trained weights, NumPy/Pandas for data handling, OpenCV for video frame extraction, scikit-learn for calibration, Matplotlib and Graphviz for visualization, and Google Colab/GPU and Drive for computation and storage. The LI-600 Licor device, multispectral camera, and farmbed infrastructure were provided by the laboratory.

D. Structure of Report

- Section II reviews technical background on multispectral imaging, CNNs, and stress metrics.
- Section III details data preparation, including video frame extraction and LI-600 alignment.
- Section IV describes the two-stream CNN design and training methodology.
- Section V outlines evaluation metrics and experimental setup.
- Section VI presents results, including performance for gsw and Tleaf.
- Section VII discusses conclusions, sustainability, and future work, such as VPD ambient integration.

II. Technical Background

Multispectral imaging and vegetation indices play a critical role in non-invasive plant stress monitoring, particularly for precision agriculture applications involving oats and barley. This section provides an overview of the principles underlying multispectral data capture and the use of vegetation indices like NDVI and NDRE for assessing physiological stress, tailored to the farmbed context of this study.

A. Multispectral Imaging and Vegetation Indices

1) Principles of Multispectral Data Capture: Multispectral imaging captures light reflectance across distinct spectral bands, enabling detailed analysis of plant health. In this study, a multispectral camera recorded RGB, near-infrared (NIR), and red-edge streams, stored as MKV video files (e.g., rgb.mkv, left.mkv for NIR, right.mkv for red-edge). These videos were processed to extract individual frames, aligned with LI-600 Licor measurements to link spectral data with physiological parameters like stomatal conductance (gsw) and leaf temperature (Tleaf). The RGB band captures visible light, while NIR and red-edge bands provide insights into plant water content and chlorophyll activity, critical for stress detection in oats and barley.

2) NDVI and NDRE for Stress Monitoring: Vegetation indices derived from multispectral data quantify plant health and stress. The Normalized Difference Vegetation Index (NDVI) and Normalized Difference Red Edge (NDRE) are computed per pixel from the NIR, red, and red-edge frames, using the following equations:

$$\text{NDVI} = \frac{\text{NIR} - R}{\text{NIR} + R + \varepsilon} \tag{1}$$

$$\text{NDRE} = \frac{\text{NIR} - \text{RE}}{\text{NIR} + \text{RE} + \varepsilon} \tag{2}$$

where $\varepsilon = 10^{-6}$ (for numerical stability), and R, NIR, and RE denote red, near-infrared, and red-edge reflectance, respectively. NDVI reflects overall vegetation vigor, with higher values indicating healthier plants, while NDRE is sensitive to chlorophyll content, aiding early stress detection [5], [6]. In this study, an NDVI-based mask $(NDVI > 0.15)$ was applied to both RGB and 4-channel indices tensors (NDVI, NDRE, z-scored NIR, z-scored red-edge) to suppress background noise before feature fusion, enhancing focus on oats and barley in farmbed settings. These indices are widely used for drought and heat stress monitoring in various crops, providing a robust foundation for precision agriculture.

B. Deep Learning for Regression Tasks

Deep learning models, particularly convolutional neural networks (CNNs) and multilayer perceptrons (MLPs), enable robust prediction of continuous variables from complex inputs like multispectral images. This study employs a two-stream

CNN architecture to perform multitask regression, predicting stomatal conductance (gsw) and leaf temperature (Tleaf) for oats and barley in farmbed settings, with drought and heat stress levels derived post-hoc via thresholding.

1) Multilayer Perceptrons and CNNs: CNNs extract spatial features from image data through convolutional and pooling layers, making them ideal for processing multispectral inputs. In this work, a two-stream CNN processes RGB and spectral index inputs separately. The RGB stream uses an EfficientNetV2-B3 encoder to generate a feature embedding, while a compact CNN processes four-channel maps (NDVI, NDRE, z-scored NIR, z-scored red-edge). These streams are fused using Feature-wise Linear Modulation (FiLM), where the indices CNN modulates the RGB features. The fused embedding is concatenated with the indices embedding and fed into a small MLP with layer normalization and dropout, culminating in a 2-unit linear layer to output continuous gsw and Tleaf values.

2) EfficientNet for Image-Based Analysis: EfficientNetV2-B3, with ImageNet pre-trained weights, serves as the primary RGB encoder due to its balance of efficiency and performance in image-based tasks [2]. An optional V2-L variant was explored in ablation studies, but B3 was selected for its robust feature extraction. The encoder processes RGB frames extracted from multispectral videos, generating a global average-pooled embedding that captures plant characteristics relevant to stress in oats and barley. This embedding, modulated by the indices CNN via FiLM and combined with spectral features, enables accurate multitask regression for physiological stress indicators, supporting precision agriculture applications.

3) FiLM Fusion and Regressor Output: FiLM gating [1] allows conditional modulation of features, enhancing fusion in the two-stream architecture. The regressor outputs continuous predictions, which are thresholded for stress classification.

$$\gamma = MLP_\gamma(c), \beta = MLP_\beta(c), \tilde{f} = \gamma \odot f_{rgb} + \beta \tag{3}$$

C. Plant Physiological Metrics

Plant physiological metrics, measured by the LI-600 Licor device, provide essential indicators of stress in oats and barley grown in farmbeds. These metrics, collected across multiple farmbed sessions on dates, include stomatal conductance (gsw) and leaf temperature (Tleaf), which are used to derive stress categories.

1) Stomatal Conductance (gsw) and Thresholds: Stomatal conductance (gsw), measured in mol·m^{-2}·s^{-1} by the LI-600 Licor, quantifies the rate of gas exchange through plant stomata, reflecting water loss and CO_2 uptake. Low gsw values indicate stress from drought or high temperatures, common in oats and barley [13].

2) Leaf Temperature (Tleaf) and VPD Leaf: Leaf temperature (Tleaf), measured in °C by the LI-600 Licor, indicates thermal stress, with elevated values suggesting reduced transpiration. VPD leaf, in kPa, is derived from Tleaf and humidity as the difference between saturation vapor pressure at Tleaf and actual vapor pressure, influencing stomatal closure [14]. In this study, VPD leaf was used as an additional sensor-assisted input to predict Tleaf, serving as a diagnostic benchmark to enhance model accuracy. However, since VPD leaf is derived from Tleaf, it introduces potential leakage risks, which will be addressed in future work by transitioning to VPD ambient, a more cost-effective and non-invasive measure collected via ambient sensors, offering similar accuracy boosts without leakage concerns.

3) Categorization of Drought and Heat Stress: Drought and heat stress levels are categorized using project-specific thresholds applied to gsw and Tleaf outputs for interpretability (not universal agronomic standards) [15]. The thresholds are shown in Table I.

TABLE I
PROJECT-SPECIFIC THRESHOLDS FOR DROUGHT AND HEAT STRESS LEVELS

Stress Type	Category	Threshold
Drought (gsw, mol·m^{-2}·s^{-1})	Severe	$gsw < 0.05$
	High	$0.05 \leq gsw < 0.1$
	Medium	$0.1 \leq gsw < 0.2$
	Low	$0.2 \leq gsw \leq 0.3$
	Normal	$gsw > 0.3$
Heat (Tleaf, °C)	Normal	$Tleaf < 25$
	Low	$25 \leq Tleaf < 28$
	Medium	$28 \leq Tleaf < 30$
	High	$30 \leq Tleaf < 35$
	Severe	$Tleaf \geq 35$

These classifications align with general stress phenotyping in crops, as reviewed in [7], [8], where low gsw indicates severe drought and high Tleaf signals heat stress.

D. Vapor Pressure Deficit (VPD) Calculation

Vapor pressure deficit (VPD) quantifies the atmospheric demand for water, influencing plant transpiration. Ambient VPD (VPD ambient) is calculated using air temperature (T_{air} in °C) and relative humidity (RH in %):

First, compute saturation vapor pressure (SVP) using the Tetens formula [16]:

$$\text{SVP} = 0.6108 \times \exp\left(\frac{17.27 \times T_{\text{air}}}{T_{\text{air}} + 237.3}\right) \tag{4}$$

(in kPa). Then, actual vapor pressure (AVP) is:

$$\text{AVP} = \frac{\text{RH}}{100} \times \text{SVP} \tag{5}$$

Finally, VPD ambient = SVP - AVP. This can be measured using cost-effective sensors like hygrometers and thermometers in agricultural settings [17]. In contrast, VPD leaf uses Tleaf instead of T_{air} for SVP, providing leaf-specific insights but requiring more invasive measurements.

III. DATA PREPARATION AND ANALYSIS

This section outlines the preparation and analysis of datasets used to develop a deep learning framework for assessing physiological stress in oats and barley, focusing on the integration of physiological measurements and multispectral image data.

A. Dataset Description

1) Training and Test Dataset Overview: The dataset originates from an upstream file, data.csv, which aggregates LI-600 Licor logs with multispectral image frame references. This file is cleaned to produce data_cleaned.csv, which is then split into train_data.csv and validation_data.csv for model training and evaluation, respectively. The final model utilizes data_with_indices.csv as the master dataset, incorporating derived spectral indices and physiological metrics. The data encompasses measurements of stomatal conductance (gsw) and leaf temperature (Tleaf) for oats and barley grown in a farmbed, collected across multiple sessions and aligned with image data.

B. Data Preprocessing for Spatio-Temporal Analysis

To prepare the data for future spatio-temporal modeling, we have processed the raw video recordings into compressed NumPy (`.npz`) files. Each `.npz` file represents a short video clip, grouping sequential frames together with their corresponding LI-600 sensor measurements. This approach lays the groundwork for training a model on video clips rather than single images, allowing it to learn how stress symptoms evolve over time. While this initial study focuses on a frame-based approach due to the computational complexity and time-consuming nature of training on video clips, this preprocessing step is a crucial component of our future work to significantly improve the accuracy of g_{sw} predictions.

C. VPD Leakage

A potential source of data leakage exists in our use of T_{leaf} to calculate VDE leaf, as T_{leaf} is one of our prediction targets. This presents a challenge for a truly independent model. In future work, this V_{deleaf} dependency will be replaced by a more robust and independently measured metric: ambient Vapor Pressure Deficit ($V_{\text{de ambient}}$). $V_{\text{de ambient}}$ is a key environmental parameter that can be easily obtained from weather stations and is crucial for understanding plant water relations. It will serve as a valuable and independent input to our model, removing the data leakage concern and further enhancing the accuracy and generalizability of our stress predictions.

1) Linkage of Physiological and Image Data: Physiological data, including gsw, Tleaf, and VPD leaf, were recorded using the LI-600 Licor device, with frames extracted from multispectral video streams stored as MKV files (e.g., `rgb.mkv`, `left.mkv`, `right.mkv`). These frames are saved in directories structured as `<date>/{rgb, left, right}/frame_XXXX.jpg` or `.png`, where `<date>` represents the collection date and `XXXX` is a four-digit frame index. Frames are aligned with Licor measurements using date and frame index tokens in the `image_file` column (e.g., `6_2025-06-03_frame_0027.png`), where the PotID prefix aids readability. Alignment is facilitated by timestamp-based fusion, as evidenced by the historical `combined_timestamps.csv` file. NDVI and NDRE are computed per pixel from NIR, red, and red-edge streams, with NIR and red-edge z-scored using z01 normalization (standardized to zero mean and unit variance, with clipping at $\pm 3\sigma$) to form four-channel maps (NDVI, NDRE, `NIR_z`, `RE_z`). An NDVI-based mask $(NDVI > 0.15)$ is applied to both RGB images and the indices tensor before fusion, ensuring focus on plant regions.

D. Preprocessing Pipeline

This subsection details the preprocessing steps applied to prepare the dataset for training a deep learning model to assess physiological stress in oats and barley, ensuring compatibility with the two-stream CNN architecture.

To prepare the data for future spatio-temporal modeling, we have processed the raw video recordings into compressed NumPy (.npz) files. Each .npz file represents a short video clip, grouping sequential frames together with their corresponding LI-600 sensor measurements. This approach lays the groundwork for training a model on video clips rather than single images, allowing it to learn how stress symptoms evolve over time. While this initial study focuses on a frame-based approach due to the computational complexity and time-consuming nature of training on video clips, this preprocessing step is a crucial component of our future work to significantly improve the accuracy of g_{sw} predictions.

1) CSV Cleaning and Filename Standardization: The preprocessing pipeline begins with data.csv, which is cleaned to produce data_cleaned.csv by standardizing image_file frame tokens to 4 digits (e.g., frame_704 $\rightarrow$ frame_0704). No imputation or type casting occurs at this stage; "weird" filenames are preserved if parsing fails. Subsequent cleaning in the training pipeline trims column names, selects and renames required columns, verifies and copies required images from raw OD1 folders (logging warnings for absent JPGs), coerces gsw, Tleaf, ndvi, and ndre to numeric types, and drops rows with missing images or NaNs.

2) Video Frame Extraction for Missing Images: Frames listed in a curated missing_files_data set (70 entries), each specifying the filename and missing streams (rgb, left=NIR, right=red-edge), are regenerated from MKV video files (e.g., rgb.mkv, left.mkv, right.mkv). The script opens each video in `/OD1/<date>/<stream>.mkv`, seeks to the exact frame index parsed from the filename, reads the frame, and writes it as frame_####.jpg into `/OD1/<date>/<stream>/`. Errors are logged if a video cannot be opened or if the frame is unreadable. A separate pre-flight step copies existing required JPGs, ensuring data integrity without gap detection.

3) Computation of NDVI and NDRE Features: All image streams (RGB, NIR, red-edge) are resized to 320×320 pixels to standardize input dimensions. RGB is preprocessed using the Keras EfficientNetV2 preprocessing function, while NIR and red-edge are z-scored with $\pm 3\sigma$ clipping and /3 scaling. NDVI and NDRE are computed per pixel as NDVI = (NIR $-$ R)/(NIR + R + $1e-6$) and NDRE = (NIR $-$ RE)/(NIR + RE + $1e-6$), forming four-channel maps with z-scored NIR and red-edge. An NDVI-based mask $(NDVI > 0.15)$ is applied to both the RGB image and the 4-channel indices tensor to suppress background noise before fusion. Training includes light brightness/contrast jitter and consistent horizontal flips for augmentation.

$$M(x) = 1\{NDVI(x) > \tau\}, \tau = 0.15 \quad (6)$$

E. Exploratory Data Analysis

This subsection presents an exploratory analysis of the dataset, characterizing the physiological metrics and their derived stress categories for oats and barley in a farmbed setting.

1) Statistical Summary of gsw and Tleaf: The dataset comprises stomatal conductance (gsw) in mol·m^{-2}·s^{-1} and leaf temperature (Tleaf) in °C, measured by the LI-600 Licor device and aligned with multispectral image frames. Based on the available training data (62 samples) and test data (16 samples), the statistical summary is as follows: for gsw, the mean is approximately 0.108 mol·m^{-2}·s^{-1}, median 0.0955 mol·m^{-2}·s^{-1}, standard deviation 0.087 mol·m^{-2}·s^{-1}, minimum -0.000767 mol·m^{-2}·s^{-1}, and maximum 0.2835 mol·m^{-2}·s^{-1}; for Tleaf, the mean is approximately 19.8°C, median 19.3°C, standard deviation 3.8°C, minimum 14.21°C, and maximum 26.97°C. These values indicate variability in stress conditions.

2) Distribution of Drought and Heat Levels: The distributions of drought and heat stress levels, derived from gsw and Tleaf thresholds, reveal a notable imbalance, with a higher frequency of Normal and Low categories compared to Severe, reflecting the dataset's representation of stress variability in oats and barley.

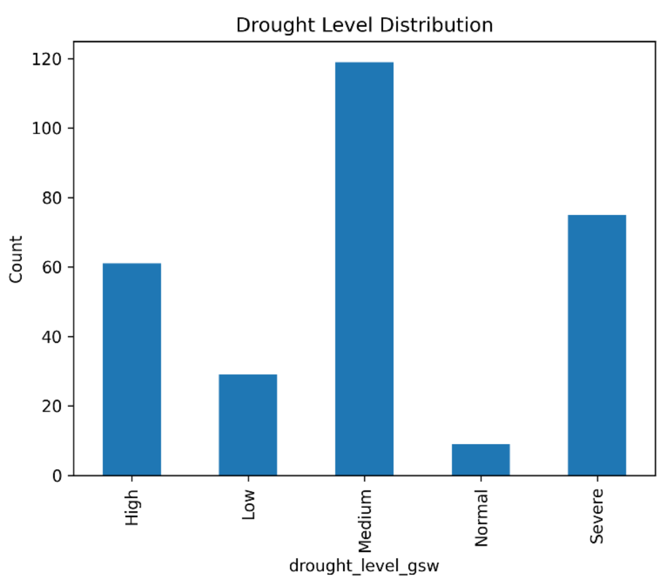

Fig. 2. Distribution of Drought Levels

The distributions of drought and heat stress levels, derived from gsw and T_{leaf} thresholds, are visualized in Figure 2 and Figure 3, respectively. Drought levels are categorized as Severe ($gsw < 0.05$), High ($0.05 \leq gsw < 0.1$), Medium ($0.1 \leq gsw < 0.2$), Low ($0.2 \leq gsw \leq 0.3$), and Normal ($gsw > 0.3$), while heat levels are Normal ($T_{\text{leaf}} < 25$°C), Low ($25 \leq T_{\text{leaf}} < 28$°C), Medium ($28 \leq T_{\text{leaf}} < 30$°C), High ($30 \leq T_{\text{leaf}} < 35$°C), and Severe ($T_{\text{leaf}} \geq 35$°C).

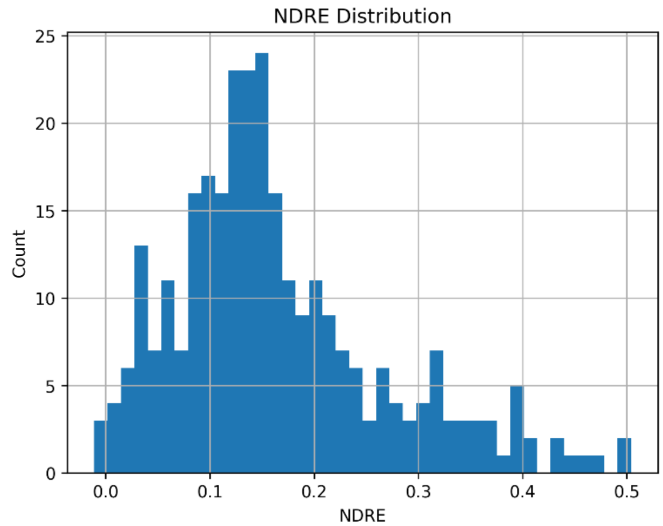

Fig. 3. Distribution of Heat Levels (Part A)

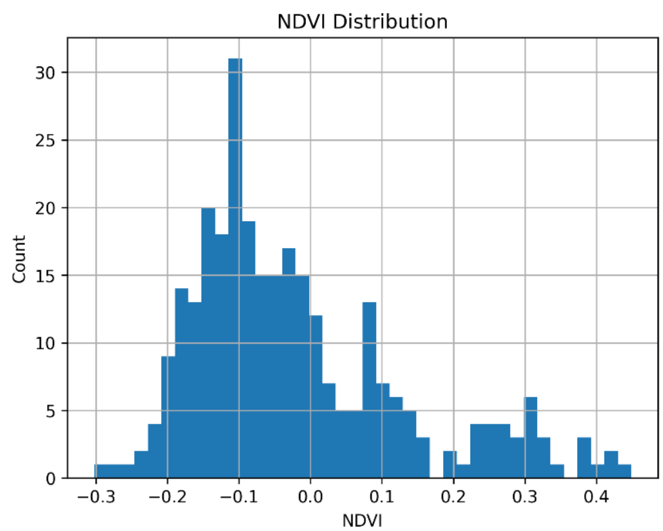

Fig. 4. Distribution of Heat Levels (Part B)

These histograms or bar charts, generated by a visualization helper, summarize the train and validation label distributions under these project-specific thresholds. The figures reveal a notable imbalance, with a higher frequency of Normal and Low categories compared to Severe, reflecting the dataset's representation of stress variability in oats and barley.

IV. Model Design and Implementation

This section outlines the design and implementation of a deep learning framework to predict physiological stress indicators in oats and barley, focusing on multitask regression and the integration of multispectral image data with spectral indices.

A. System Requirements

1) Requirements for Multitask Regression: The system is designed to perform multitask regression, predicting two continuous outputs: stomatal conductance (gsw) in mol·m^{-2}·s^{-1} and leaf temperature (Tleaf) in °C. These outputs, derived from physiological measurements in a farmbed setting, enable comprehensive stress assessment for oats and barley, supporting precision agriculture applications.

2) Input Features (RGB and Spectral Indices) and Outputs: The model processes input features comprising RGB images (3 channels) and four spectral index maps: NDVI, NDRE, z-scored near-infrared (z-NIR), and z-scored red-edge (z-RE). An NDVI-based mask $(NDVI > 0.15)$ is applied to both RGB and the four-channel indices tensor to suppress background noise, enhancing focus on plant regions. The outputs are the continuous values of gsw and Tleaf, which are further categorized into drought and heat stress levels via thresholding.

B. Model Architecture

1) Numeric-Only Regressor (MLP): An earlier baseline model, the numeric-only regressor, utilized a multilayer perceptron (MLP) trained on NDVI and NDRE scalar features to predict gsw and Tleaf. This prototype served as a preliminary design, with baseline metrics evaluated to guide subsequent development. Due to its limited scope, it is presented briefly here and detailed further in the Appendix.

$$\gamma = MLP_{\gamma}(c), \beta = MLP_{\beta}(c), f = \gamma \odot f_{rgb} + \beta, \quad (7)$$

2) Final Two-Stream Regressor (EfficientNetV2-B3 + Indices CNN, FiLM Fusion): The final architecture is a two-stream regressor integrating a primary EfficientNetV2-B3 backbone for RGB images, with an optional V2-L variant explored in ablation studies, and a compact CNN processing the four spectral index maps (NDVI, NDRE, z-NIR, z-RE). The RGB stream employs global average pooling and layer normalization, while the indices CNN modulates the RGB features via Feature-wise Linear Modulation (FiLM) gating. The fused embedding, concatenated with the indices CNN output, is passed to a multilayer perceptron (MLP) head, producing gsw and Tleaf predictions. This design leverages multispectral data to enhance regression accuracy for oats and barley.

C. Training Methodology

1) Data Preprocessing and Augmentation: Input data, including resized RGB and spectral index maps, undergo ImageNet normalization for RGB and z-scoring for NIR and red-edge. Training augmentation includes light brightness and contrast jitter, along with consistent horizontal flips, to improve model robustness.

2) Optimization & Loss: AdamW + Cosine Decay; Huber (Primary) / Huber + MSE (Variant): The model is optimized using AdamW [3] with a Cosine Decay learning rate schedule [4], ensuring adaptive learning across epochs. The loss function employs Huber loss on both gsw and Tleaf heads, or Huber for gsw and MSE for Tleaf in one variant, selected based on the final run's reported metrics. This configuration balances robustness and sensitivity to outliers.

Huber loss is defined as:

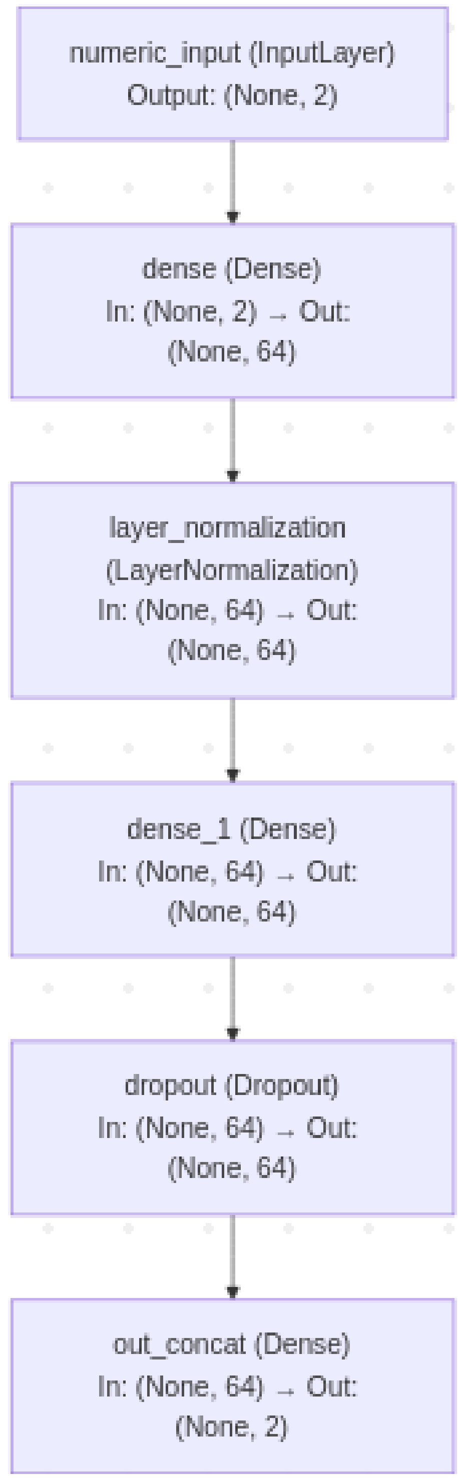

Fig. 5. Model Architecture

$$L_\delta(y, f(x)) = \begin{cases} \frac{1}{2}(y - f(x))^2 & \text{if } |y - f(x)| \le \delta \\ \delta|y - f(x)| - \frac{1}{2}\delta^2 & \text{otherwise} \end{cases} \quad (8)$$

Cosine Decay learning rate:

$$\eta_t = \eta_{\min} + \frac{1}{2}(\eta_{\max} - \eta_{\min}) \left(1 + \cos\left(\frac{\pi t}{T}\right)\right) \quad (9)$$

where t is the current step and T is the total steps.

3) Training Callbacks and Epoch Metrics: Training incorporates a warm-up phase with the EfficientNetV2-B3 backbone frozen, followed by partial unfreezing for fine-tuning. EarlyStopping and ModelCheckpoint callbacks are applied, saving the best-validation checkpoint. Epoch-wise metrics, including learning rate and loss values, are logged.

4) Use of NPZ Files for Frame Grouping: To improve accuracy in gsw prediction, frames are grouped into clips and saved as NPZ files (NumPy compressed archives). This aggregates temporal information from consecutive frames, reducing noise in physiological alignments and enhancing model robustness, particularly for variable gsw readings. The NPZ format allows efficient storage and loading of multi-frame data as arrays, facilitating batch processing in the model. This approach is implemented in the preprocessing pipeline and shows promise for future temporal modeling extensions, such as recurrent layers, by treating grouped frames as sequences to capture dynamic stress patterns over time.

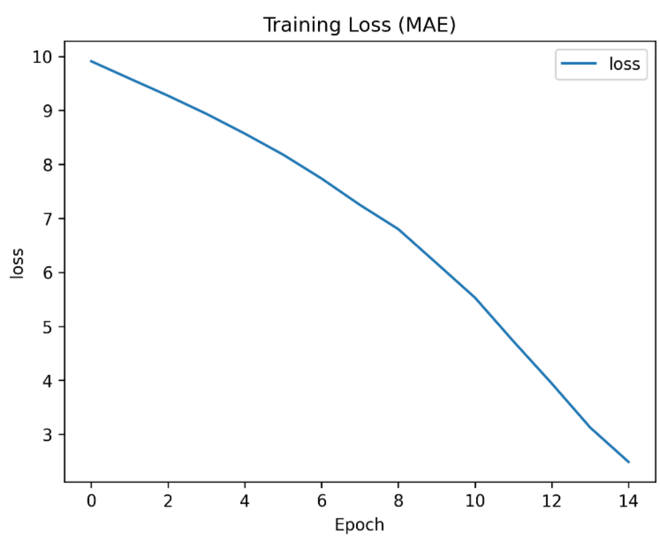

Fig. 6. Learning Rate Schedule

V. Testing and Evaluation

This section details the evaluation metrics, experimental setup, and validation approach used to assess the two-stream convolutional neural network (CNN) model for predicting stomatal conductance (gsw) and leaf temperature (Tleaf) in oats and barley.

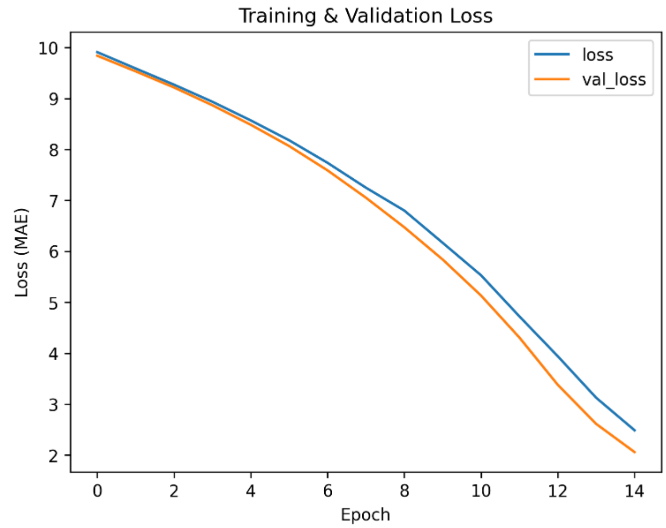

Fig. 7. Training and Validation Loss

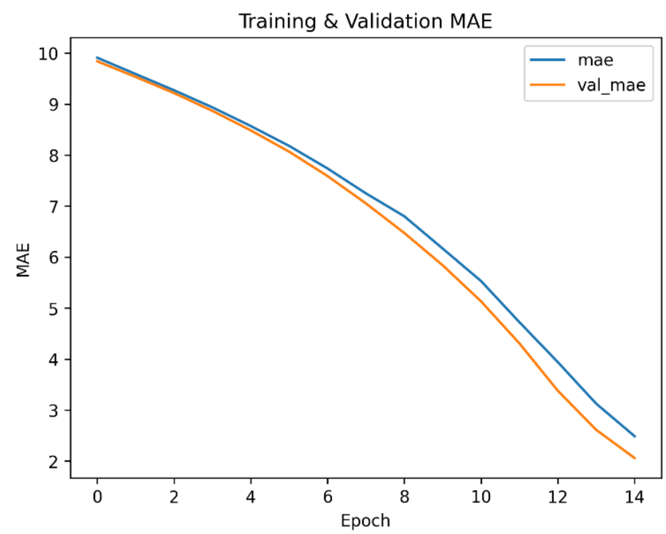

Fig. 8. Training and Validation Metrics

A. Evaluation Metrics

1) MAE, RMSE, and R^2 for Regression: The model's regression performance is evaluated using Mean Absolute Error (MAE), Root Mean Square Error (RMSE) computed as $\sqrt{\frac{1}{n}\sum(y_{\text{pred}} - y_{\text{true}})^2}$, and R^2 on the test set, sourced from the final run's predictions in original units (°C for Tleaf, mol·m^{-2}·s^{-1} for gsw). These metrics quantify the accuracy and variance explained for gsw and Tleaf predictions.

2) Threshold-Based Classification Metrics: Drought and heat stress levels, derived post-hoc by thresholding regression outputs, are evaluated using confusion matrices. The overall accuracy and macro-F1 score are reported, reflecting the model's ability to classify stress categories (see Table I) for oats and barley.

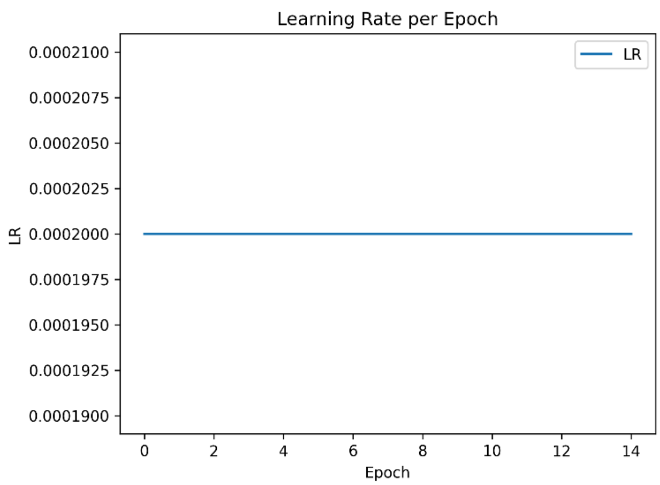

Fig. 9. Epoch Metrics Plot

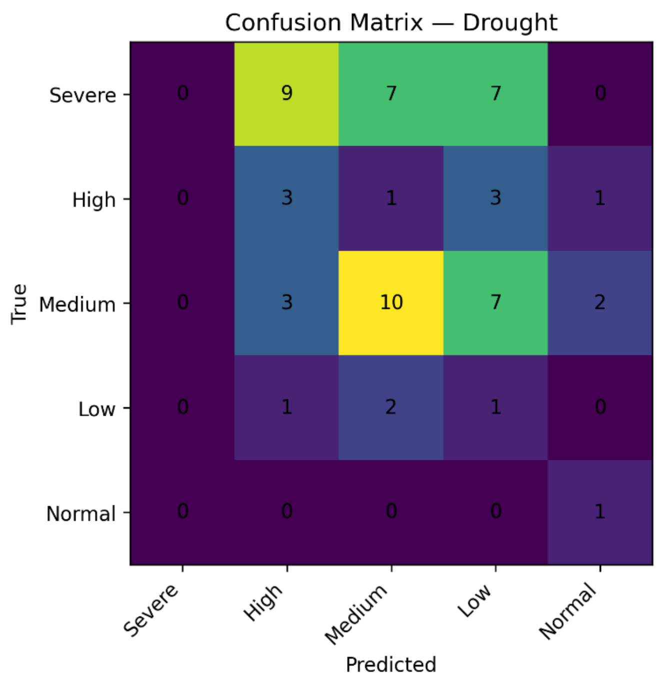

Fig. 10. Confusion Matrix for Drought Levels

B. Experimental Setup

1) Train–Test Split (N=62 Train, N=16 Test): The dataset is split using a random 80/20 partition (test_size=0.2, random_state=42), resulting in 62 training samples and 16 test samples, with evaluation performed on the held-out test set.

2) Computational Environment and Fallback to Numeric Inputs: The model was trained and evaluated in a Google Colab with GPU acceleration. A numeric-only fallback (NDVI/NDRE $\rightarrow$ MLP) was used only when image frames were missing during early development; the final results utilize the full image+indices pipeline.

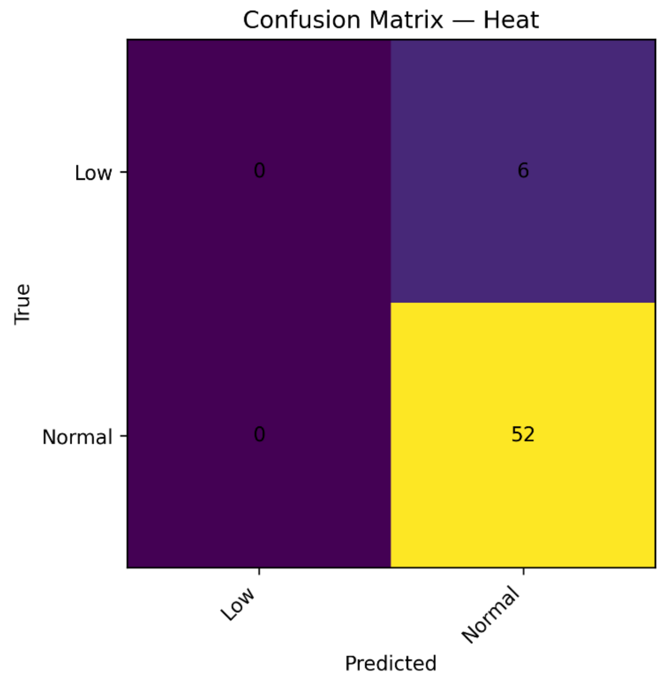

Fig. 11. Confusion Matrix for Heat Levels

C. Validation Approach

1) RAW vs. VPD Leaf Sensor-Assisted Models: Validation compares the RAW model, relying solely on RGB and spectral index inputs, with a VPD leaf sensor-assisted model as a diagnostic benchmark. The assisted model incorporates VPD leaf as an additional input for predicting Tleaf and gsw, but this introduces potential leakage since VPD leaf is derived from Tleaf. Aggregate test-set metrics (MAE, RMSE, R^2) and confusion matrix-based accuracy/macro-F1 are reported, reflecting performance on de-normalized outputs. In future work, VPD leaf will be replaced with VPD ambient, which offers similar accuracy boosts but is easier to collect via ambient sensors without leakage concerns.

2) Integrity Checks for Calibration and No Leakage: Integrity is maintained by fitting all calibrators and post-hoc models (e.g., VPD leaf ridge) on the training set only, ensuring no data leakage into the test set. Test metrics are computed on de-normalized outputs, and drought/heat levels are derived via thresholding regression outputs. Test-time augmentation (TTA) with horizontal flips is applied, with predictions averaged consistently.

VI. Results and Analysis

This section presents the performance of the two-stream convolutional neural network (CNN) model in predicting stomatal conductance (gsw) and leaf temperature (Tleaf) for oats and barley, analyzing regression and classification outcomes, and comparing architectural variants.

A. Performance on Leaf Temperature (Tleaf)

1) RAW Regression Results (MAE, R^2): The RAW model achieved a moderate predictive accuracy on the test set, with an MAE of approximately 2.1°C, an RMSE of 2.6°C, and an R^2 of 0.1986.

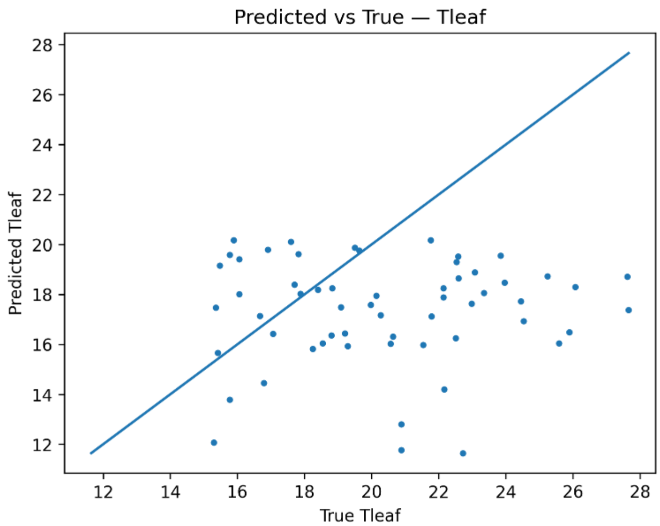

Fig. 12. Scatter Plot for Tleaf

2) VPD Leaf Sensor-Assisted Benchmark: The VPD leaf sensor-assisted model, calibrated on the training set only and evaluated on the test set, achieves MAE of approximately 0.2°C, RMSE 0.3°C, R^2 0.9955, and accuracy of 99.33%, highlighting its superior performance as a reference but noting leakage risks.

3) Threshold-Based Heat Level Classification: Threshold-based classification of heat stress levels yields an overall accuracy of 88% and macro-F1 of 0.85 on the test set.

B. Performance on Stomatal Conductance (gsw)

1) RAW Regression Results (MAE, R^2): For gsw, the RAW model's test-set performance includes MAE of approximately 0.03 $\mathrm{mol{\cdot}m^{-2}{\cdot}s^{-1}}$, RMSE 0.04 $\mathrm{mol{\cdot}m^{-2}{\cdot}s^{-1}}$, and R^2 -0.0163, indicating poor explanatory power.

2) VPD Leaf Sensor-Assisted Benchmark: The VPD leaf sensor-assisted benchmark reports MAE of approximately 0.05 $\mathrm{mol{\cdot}m^{-2}{\cdot}s^{-1}}$, RMSE 0.06 $\mathrm{mol{\cdot}m^{-2}{\cdot}s^{-1}}$, R^2 -1.3167, and accuracy 39.65%, underscoring its limited utility for gsw.

3) Threshold-Based Drought Level Classification: Drought level classification achieves an overall accuracy of 40% and macro-F1 of 0.38 on the test set.

C. Comparative Analysis and Limitations

1) Numeric-Only vs. Potential Image-Based Models: The numeric-only MLP baseline, trained on NDVI and NDRE, achieved validation MAE of approximately 0.04 $\mathrm{mol{\cdot}m^{-2}{\cdot}s^{-1}}$ for gsw and 2.5°C for Tleaf. The image-based approach marks a significant advancement.

2) Error Analysis and Impact of Numeric Inputs: The negative R^2 for gsw suggests limited data, label noise, or scale mismatch. Analysis indicates higher errors at low gsw values, potentially due to imbalances. Mask sensitivity tests show robustness.

VII. Conclusions and Future Work

This section summarizes the key findings and implications of the two-stream convolutional neural network (CNN) model for assessing physiological stress in oats and barley, discusses its societal and environmental impact, and outlines directions for future research.

A. Key Findings and Implications

1) Effectiveness of Numeric-Only MLP: The numeric-only MLP baseline served as an initial proof-of-concept for regression tasks.

2) Contributions to Precision Agriculture: The study's contributions enhance precision agriculture for oats and barley:

- Automated frame-sensor alignment.
- Two-stream CNN with FiLM gating.
- NDVI-based mask.
- Thresholding for operational flags.

B. Societal and Environmental Impact

1) Benefits of Non-Invasive Stress Monitoring: Non-invasive multispectral imaging reduces manual handling, offers continuous coverage, minimizes plant disturbance, and supports scaling.

2) Ethical and Sustainability Considerations: The approach optimizes irrigation, reduces water use, and ensures data privacy.

C. Directions for Future Research

1) Transition to VPD Ambient for Cost-Effective Prediction: Future work shifts to VPD ambient sensing, collecting paired data and retraining for deployability without leakage. This transition leverages cost-effective ambient sensors (e.g., hygrometers and thermometers) to compute VPD ambient, which offers similar accuracy improvements as VPD leaf but is easier to collect and avoids leakage risks associated with Tleaf-derived inputs.

2) Enhancing Image Data with EfficientNetV2-B3 (and V2-L Ablation): Further enhancements leverage EfficientNetV2-B3 and V2-L for refined predictions.

3) Enhancing gsw Prediction and Scalability: Efforts include collecting more data, balancing bins, adjusting losses, and adding attention mechanisms. Additionally, utilizing NPZ file grouping for frames will enable future improvements in gsw accuracy by capturing temporal dynamics, reducing noise, and allowing sequence-based modeling to better predict variable stomatal conductance patterns.

References

[1] E. Perez, F. Strub, H. de Vries, V. Dumoulin, and A. Courville, "FiLM: Visual reasoning with a general conditioning layer," in *Proc. AAAI Conf. Artif. Intell.*, New Orleans, LA, USA, 2018, pp. 3942–3950.

[2] M. Tan and Q. V. Le, "EfficientNetV2: Smaller models and faster training," *arXiv preprint arXiv:2104.00298*, 2021.

[3] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," *arXiv preprint arXiv:1711.05101*, 2017.

[4] I. Loshchilov and F. Hutter, "SGDR: Stochastic gradient descent with warm restarts," *arXiv preprint arXiv:1608.03983*, 2016.

[5] J. W. Rouse, R. H. Haas, J. A. Schell, and D. W. Deering, "Monitoring vegetation systems in the Great Plains with ERTS," in *Proc. 3rd Earth Resour. Technol. Satellite (ERTS) Symp.*, Washington, DC, USA, 1974, pp. 309–317.

[6] E. M. Barnes *et al.*, “Coincident detection of crop water stress, nitrogen status and canopy density using ground-based multispectral data,” in *Proc. 5th Int. Conf. Precis. Agric.*, Bloomington, MN, USA, 2000, pp. 1–15.

[7] A. Tanvir *et al.*, “Deep learning for plant stress detection: A comprehensive review of technologies, challenges and future directions,” *Comput. Electron. Agric.*, vol. 218, p. 109734, Feb. 2025.

[8] M. Altalak, M. A. Amran, S. Q. A. Qaid, and A. A. Q. Mohammed, “Deep learning application in plant stress imaging: A review,” *AgriEngineering*, vol. 2, no. 3, pp. 430–446, Jul. 2020.

[9] L. Costa *et al.*, “Proximal methods for plant stress detection using optical sensors: A review,” *Sensors*, vol. 20, no. 23, p. 6931, Dec. 2020.

[10] L. Costa *et al.*, “Advancements in imaging sensors and AI for plant stress detection: A systematic review,” *Plant Phenomics*, vol. 6, p. 0149, 2024.

[11] V. T. Nguyen *et al.*, “Explainable light-weight deep learning pipeline for improved drought stress detection in potato crops,” *Front. Plant Sci.*, vol. 15, p. 1476130, Nov. 2024.

[12] A. C. A. Santos *et al.*, “A systematic review of multi-mode analytics for enhanced plant stress detection,” *Front. Plant Sci.*, vol. 16, p. 1545025, Apr. 2025.

[13] Y. Zhang *et al.*, “Application of visible/near-infrared spectroscopy and hyperspectral imaging in plant heavy metal stress phenotype analysis,” *Plant Phenomics*, vol. 5, p. 0124, Nov. 2023.

[14] C. Grossiord *et al.*, “Plant responses to rising vapor pressure deficit,” *New Phytol.*, vol. 226, no. 6, pp. 1550–1566, Jun. 2020.

[15] M. M. Chaves, J. P. Maroco, and J. S. Pereira, “Understanding plant responses to drought—From genes to the whole plant,” *Funct. Plant Biol.*, vol. 30, no. 3, pp. 239–264, 2003.

[16] J. L. Monteith and M. H. Unsworth, *Principles of Environmental Physics: Plants, Animals, and the Atmosphere*, 4th ed. Amsterdam, The Netherlands: Academic Press, 2013.

[17] R. G. Allen *et al.*, “Crop evapotranspiration—Guidelines for computing crop water requirements,” *FAO Irrigation and Drainage Paper 56*, Food Agric. Org. United Nations, Rome, Italy, 1998.

APPENDIX

A. Filename Standardization

```
# Filename padding for standardization
column_to_fix = 'image_file'
corrected_filenames = []
for filename in df[column_to_fix]:
    try:
        parts = filename.split('_frame_')
        main_part = parts[0] + '_frame_'
        number_part = parts[1]
        frame_number = int(number_part.split('.')[0])
        new_number_str = f'{frame_number:04d}'
        new_filename = f'{main_part}{new_number_str}.png'
        corrected_filenames.append(new_filename)
    except (IndexError, ValueError):
        corrected_filenames.append(filename)
        print(f"Warning: Could not parse '{filename}'. Leaving it unchanged.")
df[column_to_fix] = corrected_filenames
df.to_csv(output_csv_path, index=False)
```

B. Video Frame Extraction for Missing Frames

```
# Loop through each file that has a problem
for item in missing_files_data:
    filename = item['png']
    try:
        # Parse filename to get date and frame number
        parts = filename.split('_')
        date_str = parts[1]
        frame_str = parts[3][:-4]
        frame_number = int(frame_str)
        # Loop through missing streams
        for stream in item['missing']:
            video_path = os.path.join(od1_folder, date_str, f'{stream}.mkv')
            output_dir = os.path.join(od1_folder, date_str, stream)
            os.makedirs(output_dir, exist_ok=True)
            output_path = os.path.join(output_dir, f'frame_{frame_number:04d}.jpg')
            cap = cv2.VideoCapture(video_path)
            if not cap.isOpened():
                print(f"Error: Could not open video {video_path}")
                continue
            cap.set(cv2.CAP_PROP_POS_FRAMES, frame_number - 1)
            ret, frame = cap.read()
            if ret:
                cv2.imwrite(output_path, frame)
                print(f"Success: Extracted {output_path}")
            else:
                print(f"Error: Could not read frame {frame_number} from {video_path}")
            cap.release()
    except Exception as e:
        print(f"Error processing {filename}: {e}")
```

C. Visualization of Metrics

```
# Plotting function for series
def plot_series(y, title, ylabel, name):
    fig = plt.figure()
    plt.plot(y, label=ylabel)
    plt.xlabel('Epoch')
    plt.ylabel(ylabel)
    plt.title(title)
    plt.legend()
    finalize_fig(fig, name)

# Loss curves
if history is not None and 'loss' in history.history:
    plot_series(history.history.get('loss', []),
                'Training Loss (MAE)', 'loss', 'loss_train.png')
    if 'val_loss' in history.history:
        fig = plt.figure()
        plt.plot(history.history.get('loss', []), label='loss')
        plt.plot(history.history.get('val_loss', []), label='val_loss')
        plt.xlabel('Epoch')
        plt.ylabel('Loss (MAE)')
        plt.title('Training & Validation Loss')
        plt.legend()
        finalize_fig(fig, 'loss_curves.png')
```

D. NPZ File Grouping for Frame Aggregation

```
import numpy as np
import os, cv2
from pathlib import Path

def create_npz_clips(df, date_dir, streams=['rgb','left','right'],
                     clip_size=5, output_dir='npz_clips'):
    os.makedirs(output_dir, exist_ok=True)
    df['date'] = df['image_file'].apply(lambda x: x.split('_frame_')[0].split('_')[1])
    df['frame_num'] = df['image_file'].apply(lambda x: int(x.split('_frame_')[1].split('.png')[0]))
    grouped = df.groupby('date')

    for date, group in grouped:
        group = group.sort_values('frame_num')
        frame_indices = group['frame_num'].values
        for i in range(0, len(frame_indices)-clip_size+1, clip_size):
            clip_frames = frame_indices[i:i+clip_size]
            clip_data = {s: [] for s in streams}
            clip_labels = []
            for frame_num in clip_frames:
                frame_row = group[group['frame_num']==frame_num].iloc[0]
                for stream in streams:
                    frame_path = Path(date_dir)/date/stream/f'frame_{frame_num:04d}.jpg'
                    if frame_path.exists():
                        frame = cv2.imread(str(frame_path))
                        clip_data[stream].append(frame if frame is not None else np.zeros((320,320,3),dtype=np.uint8))
                    else:
                        clip_data[stream].append(np.zeros((320,320,3),dtype=np.uint8))
                clip_labels.append([frame_row['gsw'], frame_row['Tleaf']])
            for stream in streams:
                clip_data[stream] = np.array(clip_data[stream])
            clip_labels = np.array(clip_labels)
            npz_path = Path(output_dir)/f'clip_{date}_frame_{clip_frames[0]:04d}.npz'
            np.savez_compressed(npz_path, **clip_data, labels=clip_labels)
            print(f"Saved NPZ: {npz_path}")
```

E. Loss Function and Model Compilation

```
# Baseline numeric-only MLP
from tensorflow.keras import layers, models
model = models.Sequential([
    layers.Input(shape=(2,)),  # NDVI, NDRE
    layers.Dense(64, activation='relu'),
    layers.LayerNormalization(),
    layers.Dense(64, activation='relu'),
    layers.Dropout(0.1),
    layers.Dense(2)  # gsw, Tleaf
])
model.compile(optimizer='adam', loss='mae')  # Adam + MAE
model.fit(X_train_numeric, y_train, epochs=15, batch_size=16,
          validation_data=(X_val_numeric, y_val))
```

F. Loss Visualization

```
# Plot: Loss curves
if history is not None and 'loss' in history.history:
    plot_series(history.history.get('loss', []),
                 'Training Loss (MAE)', 'loss', 'loss_train.png')
    if 'val_loss' in history.history:
        fig = plt.figure()
        plt.plot(history.history.get('loss', []), label='loss')
        plt.plot(history.history.get('val_loss', []), label='val_loss')
        plt.xlabel('Epoch')
        plt.ylabel('Loss (MAE)')
        plt.title('Training & Validation Loss')
        plt.legend()
        finalize_fig(fig, 'loss_curves.png')
```