

# Superpixels and Graph Convolutional Neural Networks for Efficient Detection of Nutrient Deficiency Stress from Aerial Imagery

Saba Dadsetan  
University of Pittsburgh  
Intelinair, Inc.  
sabadad@cs.pitt.edu

David Pichler  
Intelinair, Inc.  
david.pichler@intelinair.com

David Wilson  
Intelinair, Inc.  
david@intelinair.com

Naira Hovakimyan  
University of Illinois at Urbana Champaign  
Intelinair, Inc.  
naira@intelinair.com

Jennifer Hobbs  
Intelinair, Inc.  
jennifer@intelinair.com

## Abstract

*Advances in remote sensing technology have led to the capture of massive amounts of data. Increased image resolution, more frequent revisit times, and additional spectral channels have created an explosion in the amount of data that is available to provide analyses and intelligence across domains, including agriculture. However, the processing of this data comes with a cost in terms of computation time and money, both of which must be considered when the goal of an algorithm is to provide real-time intelligence to improve efficiencies. Specifically, we seek to identify nutrient deficient areas from remotely sensed data to alert farmers to regions that require attention; detection of nutrient deficient areas is a key task in precision agriculture as farmers must quickly respond to struggling areas to protect their harvests. Past methods have focused on pixel-level classification (i.e. semantic segmentation) of the field to achieve these tasks, often using deep learning models with tens-of-millions of parameters. In contrast, we propose a much lighter graph-based method to perform node-based classification. We first use Simple Linear Iterative Cluster (SLIC) to produce superpixels across the field. Then, to perform segmentation across the non-Euclidean domain of superpixels, we leverage a Graph Convolutional Neural Network (GCN). This model has 4-orders-of-magnitude fewer parameters than a CNN model and trains in a matter of minutes.*

## 1. Introduction

Nutrient monitoring and management is a central task for farmers throughout the growing season. To prevent major losses, growers must identify and respond to crop deficien-

cies promptly while avoiding excess applications of fertilizers that are costly and may have detrimental environmental consequences. Monitoring the health of crops, potentially across hundreds of thousands of acres of fields spread over many miles, is a daunting task without the aid of accurate, timely, automated crop intelligence at scale. Fortunately, aerial imagery from remote sensing, including satellite, airplane, and UAV, combined with image processing, computer vision, and more recently, deep learning, have led to numerous advances in precision agriculture to address these challenges.

Precision agriculture has seen rapid advancements in recent years due to the quality and availability of data combined with the advances in artificial intelligence and machine learning. Many applications focus on ground-level imagery and point clouds to detect and classify disease and pests, precisely localize weeds from crop to enable automated spraying, reconstruct scenes to enable robotic harvesting, automate counting, and numerous other tasks to enable automation and improve efficiencies [7, 52, 54, 30].

Availability of more advanced remote sensing data has also made a tremendous impact on digital agriculture. Key intelligence can be extracted at scale from imagery collected via satellite, plane, or UAV. However, aerial images of any domain, pose many challenges for computer vision techniques as they tend to be massive in size, lower in resolution, and have statistics very different from those of natural images [39, 41, 15]. Imagery size is a particular challenge as high-resolution ( $<1\text{m/pixel}$ ) images are becoming more and more prevalent: a single field may be several GB in size and contain hundreds-of-millions pixels. Approaches to improve receptive field size, multi-scale understanding, long-range interactions, and global context under-

standing in deep learning models has been an active area of research, although the focus has remained largely on natural images. Some approaches include dilated convolutions, explicit multi-scale modeling, combined CNN and RNN modules, superpixels and graphical models [44, 13, 12, 35, 34]. The use of techniques specifically adapted to handle large images is important from the practical standpoint of inference speed and efficiency, in addition to performance improvements due to incorporating information from a larger portion of the image.

While pixel-level classification may be required in some scenarios, if the primary goal is to alert farmers to regions of the field exhibiting stress, classifying each pixel may be an unnecessarily fine-grained task. Instead, we propose a method based on Simple Linear Iterative Clustering (SLIC) superpixels and a Graph Convolution Neural Network (GCN) to detect regions of nutrient deficiency stress (NDS) across a high-resolution image of field in an efficient manner. We first over-segment the image into superpixels using SLIC and characterize each superpixel by a 9-element feature vector; these superpixels become the nodes of our GCN. We then construct a fully-connected graph where each superpixel is connected to every other superpixel in the image, enabling information to propagate globally. Initial weights of the edges are further determined by the similarities between the superpixels. Finally, we compare two approaches in which the final task is either classification (whether any pixels in the super-pixel contained the target class) or regression (prediction of the fraction of target containing pixels directly). Our approach effectively determines key areas of nutrient deficiency stress while using a network containing fewer than 5,000 parameters.

## 2. Related Work

### 2.1. Computer Vision and Machine Learning for Precision Agriculture

Precision agriculture is undergoing rapid transformation and advancement thanks to new sources of data and analytical techniques in computer vision. The applications of machine learning in precision and digital agriculture are widespread including plant counting, weed detection, yield forecasting, pest and disease detection, storm damage quantification, high-throughput phenotyping, among others [40, 3, 47, 6, 27, 37]. Machine learning applied to these tasks is enabling farmers greater insight into the health of their crops, allowing them to make economically and environmentally better decisions in a more timely manner.

The detection of nutrient deficiency stress is a key task for precision agriculture and has been explored extensively using traditional computer vision and deep learning approaches with both ground-level and remotely sensed images [42, 53]. A large body of work relies on the use

of hyperspectral imaging to identify stressed areas from changes in specific bands related to suppressed chlorophyll activity[53]. Using machine learning approaches like support vector machines (SVMs) and artificial neural networks (ANNs), [58] was able to improve the estimation of nitrogen nutrient levels in rice from aerial imagery.

Recently, deep learning approaches have become increasingly common for machine vision applications in precision agriculture [28]. Particularly relevant to this work, [15] used a CNN approach to segment nutrient deficient areas, among other relevant agricultural patterns, from high resolution aerial imagery. Their work sought to determine NDS at the pixel-level and relied on a heavy, DeepLabV3-based architecture with several orders of magnitude more parameters than the GCN approach explored here. Similarly, [17] used a U-Net [46] framework for detection and prediction of nutrient deficient areas; though lighter-weight than DeepLabV3, their U-Net is still significantly heavier than our proposed method. Leveraging a subset of the dataset [15] which did not include the nutrient deficiency class, the authors of [36] proposed a Multiview Self-Constructing Graph Convolutional Network to segment the high-resolution images into six (plus background) classes. This approach used a CNN as a feature extractor followed by modules of self-construction graphs and graph convolutions before fusing to generate the final output and produced competitive results on the challenge dataset [14].

### 2.2. Superpixels

Superpixels are a common primitive in complex computer vision tasks because of their ability to produce a semantically meaningful and compact representation of images. The *Simple Linear Iterative Clustering (SLIC)* [1] algorithm for producing superpixels has become one of the most prominent approaches for generating superpixels across a variety of domains including natural images[24], medical images (e.g CT and mammography) [49], and remote sensing [19, 61] because of its speed and simplicity. SLIC adapts k-means clustering to group pixels based on their similarity in color and proximity in the image.

Previously, superpixels have been combined with graphical models such as Conditional Random Fields (CRFs) [33] to enable the segmentation of images. However, while these probabilistic graphical models successfully capture the spatial coherency of the image, they can be computationally burdensome as many methods rely on determining the maximum a posteriori (MAP) estimate to find the optimal label sequence. This has led to an interest in combining superpixel methods with deep learning-based graphical methods, which will be explored in the next section.

Within the remote sensing domain, superpixels have been combined with a variety of machine learning approaches for numerous tasks. To reduce computational

time, [16] applied a Random Forest classification based on SLIC superpixels to classify objects from remote sensing data with equivalent or superior accuracy and at a faster speed than pixel-based methods. [59] used superpixels with a graphical model for semantic segmentation of remote sensing image mapping. Particularly relevant to the present work, [29] used a SLIC and Random Forest algorithm on images collected from UAVs to classify regions of a field as soil, crop, or weed.

### 2.3. Graph Neural Networks

Convolutional Neural Networks (CNNs) are commonplace in computer vision applications because their inductive bias efficiently captures the Euclidean structure of images [32]. Graph Neural Networks (GNNs), in contrast, offer a much greater degree of flexibility in the types of data which they can model [4, 23, 48]. This has led to their adoption in numerous domains including recommendation, molecular generation, and animation [8, 55, 22].

Graph Convolutional Networks (GCNs) are a further subset of GNNs which generalize the convolutional operator of CNNs over the domain of graphs and capture the dependencies between nodes. GCN approaches fall into one of two main categories: spectral [9, 31] and spatial [20, 2, 26] approaches. Spectral methods are based on spectral graph theory and define the convolution operation in the Fourier domain by computing the eigendecomposition of the graph Laplacian [9, 18]. Spatial methods define the graph convolution operation based on a node’s spatial relation in the graph as defined by its edges, and perform convolution by aggregating the neighboring nodes [43]. Since the initial introduction of these methods, numerous advances have been made to incorporate attention, generative methods, adversarial methods, reinforcement learning, recurrent methods, and other structures common in the broader deep learning framework [51, 60, 57, 56, 45].

GNNs have recently received increased attention from the remote sensing community. To perform image classification on hyperspectral data, [10] used a cross-attention mechanism and GCN. [11] used a novel Siamese Graph Convolution Network (SGCN) to enable content-based image retrieval on remote sensing data. Very relevant to the present work, [38] constructed an attention graph convolution network to perform image segmentation. Their work leveraged single-channel, lower-resolution Synthetic Aperture Radar (SAR) as input and then generated superpixels before using a GCN to segment the regions of the image into different land usage categories.

## 3. Methods

### 3.1. Data Collection and Preprocessing

High-resolution (10cm/pixel) RGB aerial imagery of corn and soybean fields in Illinois, Indiana, and Iowa was collected during the 2019 growing season (April to October) using a Wide Area Multi-Spectral System (WAMS). These images were mosaicked to create a single large (average 15k-by-15k pixels) image per field. Images underwent orthorectification using a digital elevation model of the field to produce a planimetrically correct image [21]. Each field was imaged multiple times over the course of the season, but for this analysis, we focus only on the mid-season flights in which nutrient deficiency was likely to be present in the field. A single image was selected from 317 fields during this period. These were annotated by human experts for regions of nutrient deficiency stress; quality assurance (QA) was conducted after annotation. The 317 flights were randomly separated into 223(70%) train, 47(15%) validation, and 47(15%) test.

### 3.2. Graph Generation from Superpixels

Figure 1 shows an overview of our approach and is described by Algorithm 1.

We use the SLIC algorithm to generate 400 segmented regions over the image, setting the initial compactness to 30. Because this algorithm uses k-means clustering, in some instances we need to add some neutral nodes which are only responsible to make the number of regions consistent at 400 across images.

We define the undirected graph  $\mathcal{G}$  as the set of nodes and edges given by  $\mathcal{G} = \{\chi, \mathcal{E}\}$  where

$$\chi = \{\mathbf{x}_i \in \mathbb{R}^F | i = 1, \dots, N\} \quad (1)$$

is the set of nodes with F-dimensional attributes and

$$\mathcal{E} = \{e_{ij} \in \mathbb{R}^S | \mathbf{x}_i, \mathbf{x}_j \in \chi\} \quad (2)$$

is the set of edges with S-dimensional attributes.

Each segmented superpixel region  $K_i$  becomes a single node in the subsequent graph; the region is described by the x and y coordinates of the superpixel centroid and the multidimensional RGB color histogram ( $H_i$ ) in that region. As these are narrow-band images, many of the elements in a given channel are zero, making the distribution strongly non-Gaussian. Therefore we constructed an adjusted histogram, looking at only the non-zero elements to construct our node-feature vector. From this adjusted histogram we constructed  $F = 9$  initial features: the three-element mean  $\mu_i$ , the three-element standard deviation  $\sigma_i$ , and the three-element fractional activation  $\alpha_i$ . This last feature captures the fraction of elements in each of the original channels that are non-zero.

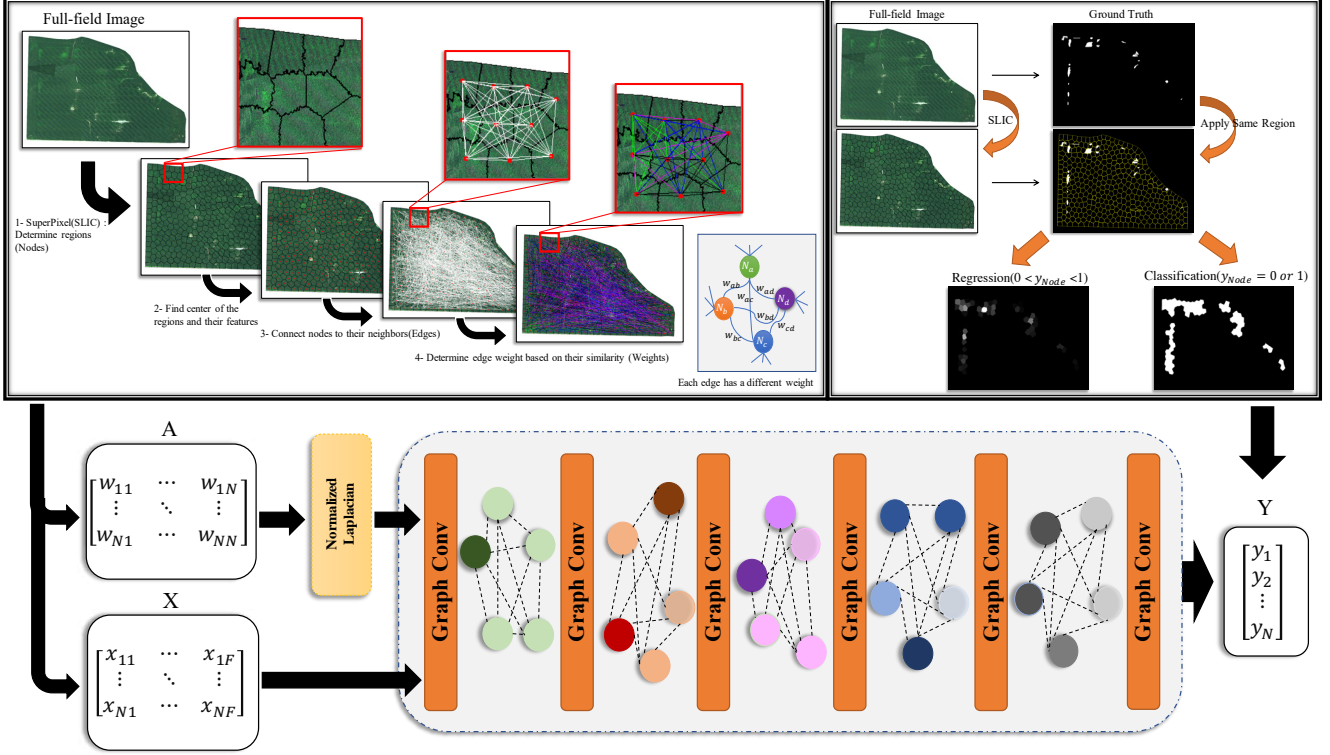


Figure 1. Our proposed method uses SLIC to first divide the image into superpixels. Top Left: A fully connected graph is constructed over the superpixels. The nodes are defined by a 9-element feature vector derived from the RGB histogram within the superpixel and the initial edge weights are computed based on the RGB histogram similarity between pairs of superpixels. Top Right: Node targets are generated for the respective regression and classification tasks. For regression, the target node value is the fraction of pixels containing the positive class (i.e. NDS present) in the superpixel. For classification, the node is labeled as 1 if *any* pixel in the superpixel contained NDS and 0 otherwise. Bottom: The initialized adjacency matrix  $A$  and node feature matrix  $X$  are fed into six graph convolutional layers to predict the final node output  $Y$ .

We connect all ( $N = 400$ ) nodes to generate a fully-connected graph and initialize the edge weights as follows. First, we compute the Bhattacharyya coefficient[5]  $BC_{ij}$  between the multi-dimensional RGB histograms,  $H_i$  and  $H_j$ , of pairs of nodes  $K_i$  and  $K_j$ .

We define the similarity between nodes as  $W_{ij} = 1 - BC_{ij}$ . As the Bhattacharyya coefficient is defined on  $[0,1]$ , this maps nodes with low overlap (i.e. high BC value) to a similarity score close to 0 and nodes with high overlap (i.e. low BC value) to a similarity score close to 1. We then set our initial edge weight to this similarity measure.

With this initialization, we represent  $\mathcal{G}$  by its adjacency matrix  $A \in \{0,1\}^{N \times N}$ , node features  $X \in \mathbb{R}^{N \times F}$ , and edge features  $E \in \mathbb{R}^{N \times N \times S}$ . Applying the *renormalization trick* according to [31], we define the convolved signal matrix  $Z \in \mathbb{R}^{N \times F}$  as  $Z = \tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}} X \theta$  where  $\theta \in \mathbb{R}^{C \times F}$  is the matrix of filter parameters for C input channels and  $\tilde{D}_{ii} = \sum_j A_{ij}$  is a layer-specific trainable matrix of weights.

### 3.3. Target Map Creation

We explore two approaches(classification and regression) using this framework and construct an appropriate target map for each as seen in the top-right sub-box of Figure 1.

For the “classification” target, the node’s label will be binary. We define the node’s label  $y_i = 1$  if *any* ground-truth pixel with label 1 is inside that node’s region, otherwise, it is considered 0. This has the effect of boosting the presence of the NDS class, which is rare. A different threshold could be used to generate the classification target which may make the task easier for the model to learn, however, we chose this approach because of the desire for the model to aggressively identify regions of stress.

In contrast, for the “regression” target, we seek to identify the fraction of pixels within a superpixel that contain NDS. We determine the label of each node  $y_i$  as a proportion of pixels with label 1 (i.e. NDS present) inside each superpixel region boundary. In this way, a number between 0 and 1 is assigned as the label of that node.



---

**Algorithm 1: Graph Generation from superpixel**

---

**Data:** Image  $I$ , Superpixel technique  $SLIC$ ,  
Number of nodes  $N$ , Task  $T$   
**Result:** Adjacency Matrix  $\mathcal{A}$ , Feature Matrix  $\mathcal{X}$ ,  
Label Vector  $\mathcal{Y}$

```
begin
    // Adjacency & Feature Matrix Creation
    Nodes : { $K_1, \dots, K_N$ }  $\leftarrow SLIC(I, N)$ 
    for  $K_i$  in Nodes do
         $\mathcal{X}_i \leftarrow \text{Feature-Extractor}(K_i)$ 
         $H_i \leftarrow \text{MultiD-Histogram}(K_i)$ 
    for  $K_i$  in Nodes do
        for  $K_j$  in Nodes do
             $BC_{ij} \leftarrow \text{Bhattacharyya-Coef}(H_i, H_j)$ 
             $A_{ij} \leftarrow 1 - BC_{ij}$ 

begin
    // Target Map Creation
    if  $T$  is "Classification" then
         $\mathcal{Y}_i \leftarrow 1$  if in  $K_i$  any pixels contain the
        target class 1
    if  $T$  is "Regression" then
         $\mathcal{Y}_i \leftarrow$  fraction of target class 1 containing
        pixels in  $K_i$ 
```

---

### 3.4. Model Architecture and Training

The above adjacency matrix  $\mathcal{A}$  and node feature matrix  $\mathcal{X}$  are input into our neural network. We use six graph convolutional layers with [320, 1056, 1056, 1056, 1056, 33] parameters in each respective layer. The first five layers have  $l2$  kernel regularization and  $elu$  activation. The last graph convolutional layer uses a *sigmoid* activation function to produce the output  $\mathcal{Y}$ .

For both tasks, the loss is computed as the Dice Loss [50] between the target and predicted maps. We elect to use Dice loss for the regression task because the value being regressed is a value between 0 and 1 corresponding to the fraction of pixels in the superpixel containing the positive class. The matrix multiplication to compute the intersection of the target and predicted maps is computed as in the classification tasks except that the predicted map is continuous instead of binary. This has the effect of "softening" the impact from nodes which have minimal NDS.

All models are trained with a batch size of 32 for 200 epochs using Adam Optimization with an initial learning rate of  $1e-3$ . This value is decayed by a factor of 10 if no improvement to the validation loss is seen for a period of 10 epochs.

The Graph Convolutional Network is build in Spektral [25] and Keras (version 2.2.5) with a Tensorflow (version 1.15) backend. Models are trained on a machine with

1 NVIDIA Tesla V100 GPUs with 32GiB memory in total.

## 4. Results

### 4.1. Classification Model

Results of the classification model are shown in Table 1. This model had a dice loss of 0.83. We then threshold the node probabilities at  $p = 0.4$  to compare the regional overlap with the target map; at this threshold the precision is 0.33, recall is 0.65, F1-score is 0.43 and IOU-score is 0.28. Note that the threshold is adjusted to 0.4 based on precision-recall curve to achieve the highest F1-score. Recall that we set the target value of the node to 1 if any pixel in that superpixel contained NDS; this is an extremely difficult task as the model may be penalized for miss-classifying a node even if it has an very minimal amount of NDS. Exploring this thresholding is the focus of future work.

Qualitative results are shown in Figure 2. We see that the model does a good job identifying key areas of nutrient deficiency stress. The first row shows a clear example of how the use of superpixels and the GCN finds the large region near the bottom of the field exhibiting NDS; while a more detailed pattern exists at the pixel level, those fine-grained details are not necessary to alert the farmer to this key problem in the field.

### 4.2. Regression Model

In creating the target nodes for the classification task, we made the decision to label the node as a positive class if any pixel in the superpixel contained NDS; this is inevitably removing information about the extent of NDS in that region. Therefore, we sought to predict the fraction of pixels labeled as nutrient deficient directly through a regression task.

The Dice loss of the regression model is 0.95 as seen in Table 1. The predictions obtained via direct regression of the fraction of NDS in a superpixel are shown in 3. These results demonstrate that regression model did not perform as well as the classification task even though a continuous target was provided. We elected to use Dice loss of both tasks for more direct comparison, however, other regression-appropriate loss functions, like RMSE, could be used instead and may improve performance; this will be explored in future work.

## 5. Discussion

### 5.1. Performance and Comparison to Other Methods

Both the classification and regression models identify key areas of nutrient deficiency stress from high-resolution RGB aerial imagery. We used an "aggressive" target creation step for the classification model, setting a node with any pixels containing NDS to 1. This threshold could be

| Model                             | Dice Loss | Precision | Recall | F1-score | IOU-score |
|-----------------------------------|-----------|-----------|--------|----------|-----------|
| Graph convolution- classification | 0.83      | 0.33      | 0.65   | 0.43     | 0.28      |
| Graph convolution- regression     | 0.95      | *         | *      | *        | *         |

Table 1. Results from the GCN-based classification and regression models on the test set.

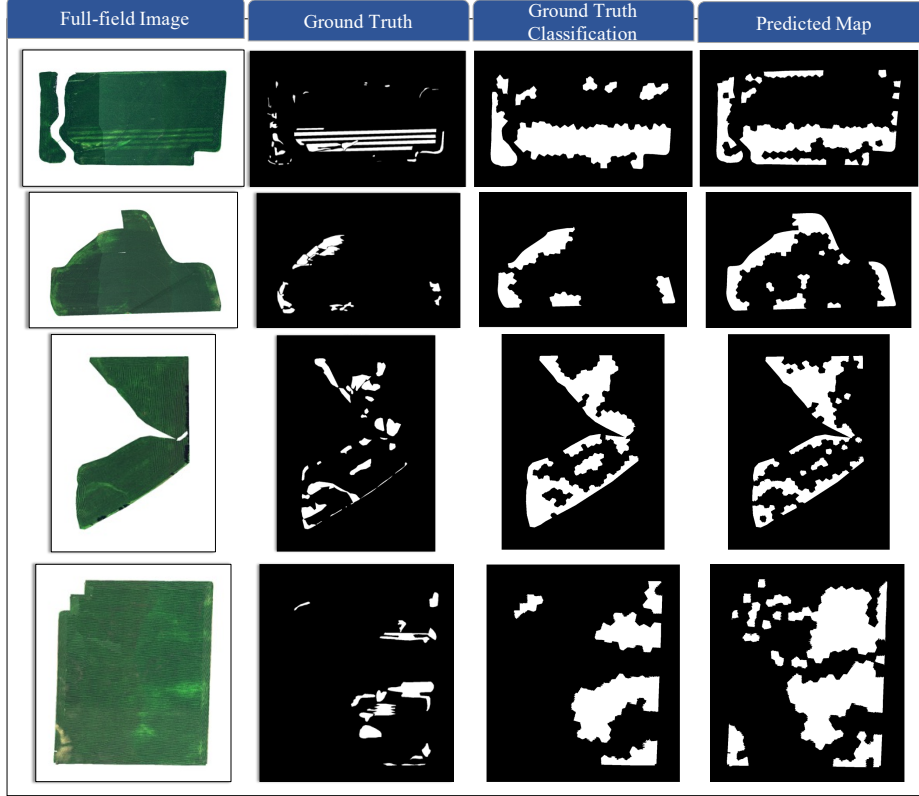


Figure 2. Example results for the classification model. The Ground Truth map shows pixel-level annotations and while the Ground Truth Classification and Predicted Map show node-level labels at the superpixel level.

tuned as a hyperparameter to alter the model’s output depending on the desired balance between precision and recall; a value which produces the highest F1 score may be chosen, alternately there may be reasons to value higher precision at the expense of a lower recall. Although the regression model did not perform as well, we believe further exploration around different loss functions may lead to enhanced performance.

Detection of nutrient deficiency stress from remotely sensed data is not a new task with past approaches spanning traditional image processing as well as more modern deep learning methods. Recent deep learning approaches have performed well on the task, although they have been largely focused on pixel-level classification. The DeepLabv3+ multi-class models of [15] demonstrated an IOU on the nutrient deficiency class in the range 0.3418 - 0.3940 (validation) and 0.3937 - 0.4415 (test) on their dataset. In [17], the authors achieved a top single-image

IOU of 0.34 and F1 Score of 0.43 using a U-Net with EfficientNet backbone. Because our classification is at the node (i.e. superpixel) level and uses only RGB (no NIR channel), it is not directly comparable to these approaches. We believe that incorporating additional channels and vegetative indices within the node features will further improve the models framework.

Additionally, our model with only 4,577 parameters is orders of magnitude smaller than either of these approaches, each which has tens-of-millions of parameters. Both the classification and regression models trained in a matter of minutes: over 200 epochs could be run on readily available hardware in around 5 minutes and GCN inference can run at 650frames/sec. Pixel-level accuracy may be a requirement for certain tasks, however, in the regime in which identification of regions is sufficient, our approach offers tremendous advantages on the computational front. As the amount of data grows through additional channels (e.g. hyperspectral

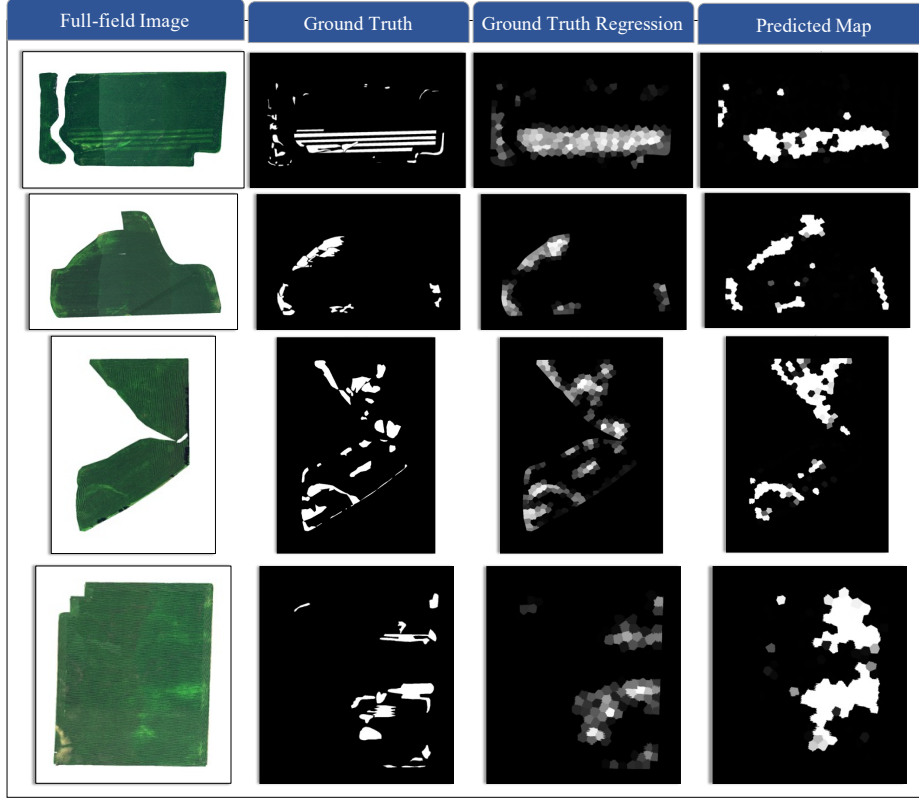


Figure 3. Example results for the regression model. The Ground Truth map shows pixel-level annotations and while the Ground Truth Regression and Predicted Map show node-level labels at the superpixel level.

imagery) or revisits (i.e. longitudinal imagery) this can become further computationally burdensome to process. As the goal of such a model is to provide farmers with intelligence about their fields in order to improve economic efficiencies while minimizing any negative environmental impact, such considerations around model design are important.

## 5.2. Future Work

For our node features, we focus only on 9 simple features which characterized the RGB histogram. Future work will explore the use of more complex features including additional moments of the distribution as well as additional channels (like NIR) and indices (like NDVI, GNDVI). The GCN framework enables the addition of other relevant features such as these without dramatically increasing the size or complexity of the network.

Similarly, we have chosen to focus on only a single (binary) task for this analysis. However, this method can easily be extended to other patterns of interest such as flooding, inhibited emergence, drydown, weeds, etc..

Furthermore we have focus only on a single point in time for this analysis. [17] demonstrated the significant boost in performance which can be achieved through the use of

longitudinal data. The superpixel-GCN framework is well posed to handle longitudinal data. A key advantage of this framework is the ability to handle registration shifts which may occur across multiple images. More specifically, perfect pixel-level alignment is not guaranteed across the sequence of images. While CNN-based approaches have been successful even despite this noise, the GCN approach offers additional flexibility as the graph can be constructed to enable information flow across all regions of all images.

Finally, in this work we have used a very simple GCN to demonstrate the usefulness and effectiveness of this approach. Additional modules such as attention [51] have proven to be highly successful in both natural image and remote sensing domains [10] and are the focus of future work.

## 5.3. Conclusion

In this work we have presented an approach which uses a Graph Convolutional Neural Network on top of SLIC Superpixels to efficiently detect regions of nutrient deficiency stress. Our model identifies regions of the field experiencing nutrient deficiency stress and has only 4,577 parameters. This work has only used a limited set of node features and uses only RGB imagery, however, we believe the performance of this approach demonstrates its usefulness and via-

bility to detect NDS and other relevant agricultural patterns from large remotely sensed imagery in a computationally efficient manner.

## References

- [1] Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Süsstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *IEEE transactions on pattern analysis and machine intelligence*, 34(11):2274–2282, 2012. 2
- [2] James Atwood and Don Towsley. Diffusion-convolutional neural networks. *arXiv preprint arXiv:1511.02136*, 2015. 3
- [3] Alexandre Barbosa, Rodrigo Trevisan, Naira Hovakimyan, and Nicolas F Martin. Modeling yield response to crop management using convolutional neural networks. *Computers and Electronics in Agriculture*, 170:105197, 2020. 2
- [4] Peter W. Battaglia, Jessica B. Hamrick, Victor Bapst, Alvaro Sanchez-Gonzalez, Vinícius Flores Zambaldi, Mateusz Malinowski, Andrea Tacchetti, David Raposo, Adam Santoro, Ryan Faulkner, Çağlar Gülçehre, H. Francis Song, Andrew J. Ballard, Justin Gilmer, George E. Dahl, Ashish Vaswani, Kelsey R. Allen, Charles Nash, Victoria Langston, Chris Dyer, Nicolas Heess, Daan Wierstra, Pushmeet Kohli, Matthew Botvinick, Oriol Vinyals, Yujia Li, and Razvan Pascanu. Relational inductive biases, deep learning, and graph networks. *CoRR*, abs/1806.01261, 2018. 3
- [5] Anil Bhattacharyya. On a measure of divergence between two statistical populations defined by their probability distributions. *Bull. Calcutta Math. Soc.*, 35:99–109, 1943. 4
- [6] Justine Boulent, Samuel Foucher, Jérôme Théau, and Pierre-Luc St-Charles. Convolutional neural networks for the automatic identification of plant diseases. *Frontiers in Plant Science*, 10:941, 2019. 2
- [7] Justine Boulent, Pierre-Luc St-Charles, Samuel Foucher, and Jérôme Théau. Automatic detection of flavescence dorée symptoms across white grapevine varieties using deep learning. *Frontiers in Artificial Intelligence*, 3:96, 2020. 1
- [8] Michael M Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, and Pierre Vandergheynst. Geometric deep learning: going beyond euclidean data. *IEEE Signal Processing Magazine*, 34(4):18–42, 2017. 3
- [9] Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann LeCun. Spectral networks and locally connected networks on graphs. *arXiv preprint arXiv:1312.6203*, 2013. 3
- [10] Weiwei Cai and Zhanguo Wei. Remote sensing image classification based on a cross-attention mechanism and graph convolution. *IEEE Geoscience and Remote Sensing Letters*, 2020. 3, 7
- [11] Ushasi Chaudhuri, Biplob Banerjee, and Avik Bhattacharya. Siamese graph convolutional network for content based remote sensing image retrieval. *Computer Vision and Image Understanding*, 184:22–30, 2019. 3
- [12] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017. 2
- [13] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018. 2
- [14] Mang Tik Chiu, Xingqian Xu, Kai Wang, Jennifer Hobbs, Naira Hovakimyan, Thomas S. Huang, and Honghui Shi. The 1st agriculture-vision challenge: Methods and results. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2020. 2
- [15] Mang Tik Chiu, Xingqian Xu, Yunchao Wei, Zilong Huang, Alexander G Schwing, Robert Brunner, Hrant Khachatrian, Hovnatan Karapetyan, Ivan Dozier, Greg Rose, et al. Agriculture-vision: A large aerial image database for agricultural pattern analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2828–2838, 2020. 1, 2, 6
- [16] Ovidiu Csillik. Fast segmentation and classification of very high resolution remote sensing data using slic superpixels. *Remote Sensing*, 9(3):243, 2017. 3
- [17] Saba Dadsetan, Gisele Rose, Naira Hovakimyan, and Jennifer Hobbs. Detection and prediction of nutrient deficiency stress using longitudinal aerial imagery, 2020. 2, 6, 7
- [18] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. *arXiv preprint arXiv:1606.09375*, 2016. 3
- [19] Dawa Derksen, Jordi Inglada, and Julien Michel. Scaling up slic superpixels using a tile-based approach. *IEEE Transactions on Geoscience and Remote Sensing*, 57(5):3073–3085, 2019. 2
- [20] David Duvenaud, Dougal Maclaurin, Jorge Aguilera-Iparraguirre, Rafael Gómez-Bombarelli, Timothy Hirzel, Alán Aspuru-Guzik, and Ryan P Adams. Convolutional networks on graphs for learning molecular fingerprints. *arXiv preprint arXiv:1509.09292*, 2015. 3
- [21] Feng Gao, Jeffrey G Masek, and Robert E Wolfe. Automated registration and orthorectification package for landsat and landsat-like data processing. *Journal of Applied Remote Sensing*, 3(1):033515, 2009. 3
- [22] Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. In *International Conference on Machine Learning*, pages 1263–1272. PMLR, 2017. 3
- [23] Marco Gori, Gabriele Monfardini, and Franco Scarselli. A new model for learning in graph domains. In *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005.*, volume 2, pages 729–734. IEEE, 2005. 3
- [24] Stephen Gould, Jim Rodgers, David Cohen, Gal Elidan, and Daphne Koller. Multi-class segmentation with relative location prior. *International Journal of Computer Vision*, 80(3):300–316, 2008. 2
- [25] Daniele Grattarola and Cesare Alippi. Graph neural networks in tensorflow and keras with spektral. *arXiv preprint arXiv:2006.12138*, 2020. 5



- [26] William L Hamilton, Rex Ying, and Jure Leskovec. Inductive representation learning on large graphs. *arXiv preprint arXiv:1706.02216*, 2017. 3
- [27] Mehdi Hosseini, Hannah R Kerner, Ritvik Sahajpal, Estefania Puricelli, Yu-Hsiang Lu, Afolarin Fahd Lawal, Michael L Humber, Mary Mitkish, Seth Meyer, and Inbal Becker-Reshef. Evaluating the impact of the 2020 iowa derecho on corn and soybean fields using synthetic aperture radar. *Remote Sensing*, 12(23):3878, 2020. 2
- [28] Andreas Kamilaris and Francesc X Prenafeta-Boldú. Deep learning in agriculture: A survey. *Computers and electronics in agriculture*, 147:70–90, 2018. 2
- [29] Kensuke Kawamura, Hidetoshi Asai, Taisuke Yasuda, Phenphit Soisouvanh, and Sengthong Phongchanmixay. Discriminating crops/weeds in an upland rice field from uav images with the slic-rf algorithm. *Plant Production Science*, pages 1–18, 2020. 3
- [30] Saeed Khaki, Hieu Pham, Ye Han, Andy Kuhl, Wade Kent, and Lizhi Wang. Convolutional neural networks for image-based corn kernel detection and counting. *Sensors*, 20(9), 2020. 1
- [31] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*, 2017. 3, 4
- [32] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25:1097–1105, 2012. 3
- [33] John Lafferty, Andrew McCallum, and Fernando CN Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. 2001. 2
- [34] Di Lin, Yuanfeng Ji, Dani Lischinski, Daniel Cohen-Or, and Hui Huang. Multi-scale context intertwining for semantic segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018. 2
- [35] Guosheng Lin, Anton Milan, Chunhua Shen, and Ian Reid. Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1925–1934, 2017. 2
- [36] Qinghui Liu, Michael C. Kampffmeyer, Robert Jenssen, and Arnt-Borre Salberg. Multi-view self-constructing graph convolutional networks with adaptive class weighting loss for semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2020. 2
- [37] B. Lyu, S. D. Smith, and K. A. Cherkauer. Fine-grained recognition in high-throughput phenotyping. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 320–329, 2020. 2
- [38] Fei Ma, Fei Gao, Jinping Sun, Huiyu Zhou, and Amir Hussain. Attention graph convolution network for image segmentation in big sar imagery data. *Remote Sensing*, 11(21):2586, 2019. 3
- [39] Lei Ma, Yu Liu, Xueliang Zhang, Yuanxin Ye, Gaofei Yin, and Brian Alan Johnson. Deep learning in remote sensing applications: A meta-analysis and review. *ISPRS Journal of Photogrammetry and Remote Sensing*, 152:166–177, 2019. 1
- [40] Lonesome Malambo, Sorin Popescu, Nian-Wei Ku, William Rooney, Tan Zhou, and Samuel Moore. A deep learning semantic segmentation-based approach for field-level sorghum panicle counting. *Remote Sensing*, 11(24):2939, 2019. 2
- [41] Dimitrios Marmanis, Mihai Datcu, Thomas Esch, and Uwe Stilla. Deep learning earth observation classification using imagenet pretrained networks. *IEEE Geoscience and Remote Sensing Letters*, 13(1):105–109, 2015. 1
- [42] CY Mee, SK Balasundram, and AHM Hanif. Detecting and monitoring plant nutrient stress using remote sensing approaches: A review. *Asian J. Plant Sci*, 16(1), 2017. 2
- [43] Federico Monti, Davide Boscai, Jonathan Masci, Emanuele Rodola, Jan Svoboda, and Michael M Bronstein. Geometric deep learning on graphs and manifolds using mixture model cnns. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5115–5124, 2017. 3
- [44] Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*, 2016. 2
- [45] Xiaojuan Qi, Renjie Liao, Jiaya Jia, Sanja Fidler, and Raquel Urtasun. 3d graph neural networks for rgb-d semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5199–5208, 2017. 3
- [46] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 2
- [47] Inkyu Sa, Zetao Chen, Marija Popović, Raghav Khanna, Frank Liebisch, Juan Nieto, and Roland Siegwart. weed-net: Dense semantic weed classification using multispectral images and mav for smart farming. *IEEE Robotics and Automation Letters*, 3(1):588–595, 2017. 2
- [48] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. The graph neural network model. *IEEE Transactions on Neural Networks*, 20(1):61–80, 2008. 3
- [49] Jaime Simarro, Zohaib Salahuddin, Ahmed Gouda, and Anindo Saha. Leveraging slic superpixel segmentation and cascaded ensemble svm for fully automated mass detection in mammograms. *arXiv preprint arXiv:2010.10340*, 2020. 2
- [50] Carole H Sudre, Wenqi Li, Tom Vercauteren, Sebastien Ourselin, and M Jorge Cardoso. Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations. In *Deep learning in medical image analysis and multimodal learning for clinical decision support*, pages 240–248. Springer, 2017. 5
- [51] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017. 3, 7

- [52] Aichen Wang, Wen Zhang, and Xinhua Wei. A review on weed detection using ground-based machine vision and image processing techniques. *Computers and electronics in agriculture*, 158:226–240, 2019. 1
- [53] Marek Wójtowicz, Andrzej Wójtowicz, Jan Piekarczyk, et al. Application of remote sensing methods in agriculture. *Communications in Biometry and Crop Science*, 11(1):31–50, 2016. 2
- [54] F. Yandun, A. Silwal, and G. Kantor. Visual 3d reconstruction and dynamic simulation of fruit trees for robotic manipulation. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 238–247, 2020. 1
- [55] Rex Ying, Ruining He, Kaifeng Chen, Pong Eksombatchai, William L Hamilton, and Jure Leskovec. Graph convolutional neural networks for web-scale recommender systems. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 974–983, 2018. 3
- [56] Jiaxuan You, Rex Ying, Xiang Ren, William Hamilton, and Jure Leskovec. Graphrnn: Generating realistic graphs with deep auto-regressive models. In *International Conference on Machine Learning*, pages 5708–5717. PMLR, 2018. 3
- [57] Wenchao Yu, Cheng Zheng, Wei Cheng, Charu C Aggarwal, Dongjin Song, Bo Zong, Haifeng Chen, and Wei Wang. Learning deep network representations with adversarially regularized autoencoders. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2663–2671, 2018. 3
- [58] Hainie Zha, Yuxin Miao, Tiantian Wang, Yue Li, Jing Zhang, Weichao Sun, Zhengqi Feng, and Krzysztof Kusnierek. Improving unmanned aerial vehicle remote sensing-based rice nitrogen nutrition index prediction with machine learning. *Remote Sensing*, 12(2):215, 2020. 2
- [59] Guanyun Zhang, Xiuping Jia, and Jiankun Hu. Superpixel-based graphical model for remote sensing image mapping. *IEEE Transactions on Geoscience and Remote Sensing*, 53(11):5861–5871, 2015. 3
- [60] Ziwei Zhang, Peng Cui, and Wenwu Zhu. Deep learning on graphs: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 2020. 3
- [61] Baokai Zu, Kewen Xia, Tiejun Li, Ziping He, Yafang Li, Jingzhong Hou, and Wei Du. Slic superpixel-based l2, l1-norm robust principal component analysis for hyperspectral image classification. *Sensors*, 19(3):479, 2019. 2