HIdentifier: A Method in Agriculture CPS Framework to Automatically Identify Disease Hotspots Using Message Passing in Graph

Kiran K. Kethineni*, Saraju P. Mohanty* and Elias Kougianos†
*Department of Computer Science and Engineering, University of North Texas, USA
†Department of Electrical Engineering, University of North Texas, USA
Email: kirankumar.kethineni@unt.edu, saraju.mohanty@unt.edu, elias.kougianos@unt.edu

Abstract—Plant diseases affect the health and yield of the crop which can in turn impact farmers economically and pose a threat to food security across the globe. To protect crops from disease and obtain high yield, many Agriculture Cyber-Physical Systems (A-CPS) have been developed with the ability to detect the presence of a disease. In addition, to effectively control the spread of the disease and reduce its impact on the crops, insights on the spread of the disease and locations of disease hotspots that act as sources for the disease must be identified. HIdentifier proposes a novel graph-based spatial analysis method that takes segmented images of the diseased farmland as input. All the identified disease segments in the input image are represented as a graph to capture the spatial relation between diseased locations. Message passing is performed over the graph to know about the neighbors of each node and effectively identify hotspots. The solution proposed has been validated on images of leaves with disease spots on them annotated and the results were similar to that of visual estimation by humans. These results can be instrumental in automating disease control with the help of Internet-of-Agro-Things (IoAT) for effective disease management.

Keywords—Smart Agriculture; Agriculture Cyber-Physical System (A-CPS); Internet-of-Agro-Things (IoAT); Artificial Intelligence; Graphs; Message Passing.

I. INTRODUCTION

Agriculture plays a crucial role in the development and sustainability of humans by serving as the primary source of food. In addition, helps in the socio-economic development of rural communities with employment and economic generation. Any change in the physiological process of a plant affecting its growth and functionality can be considered a plant disease [1]. Interaction between plants, plant diseases and various factors like environment, soil, pathogens, genetics, insects and pests that can spread the disease is represented by the Plant disease triangle in Figure 1. The study focused on the causes, detection, management of plant diseases and their impacts on agriculture is Plant Pathology. With rising populations across the world and plants constituting more than 80% of the diet of humans, plant diseases pose a threat to food security worldwide [2]. Plant diseases not only affect the availability of food but also affect the quality of food being produced which in turn can deteriorate the health of those people who have access to the food. Thus, effective plant disease management is essential to make sure every human has access to quality food to meet his dietary needs [3].
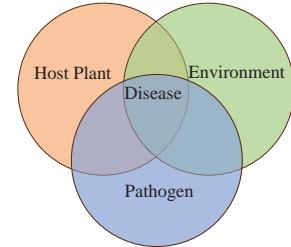

Figure 1: Plant disease triangle.

To better handle the propagation of disease in farmland, the presence of the disease must be detected. Traditionally humans would have to scout through the farmland inspecting plants individually. But in cases where the farmland is large (e.g. hundreds of acres), manual scouting is not feasible. Advancements in sensor development and communication technologies resulted in agriculture-specific Internet-of-Agro-Things (IoAT) or agriculture Cyber-Physical Systems (A-CPS) [4], [5]. These helped in developing Smart Agriculture solutions by incorporating sensors to monitor various aspects of plants [6]. In addition, Artificial Intelligence (AI) and Machine learning (ML) have been integrated with IoAT in smart agriculture to automate agricultural processes. These methods can also help us understand the spatiotemporal spread of the disease to limit the spread and reduce the damage [7]. Likewise, the current article "HIdentifier" presents a spatial analysis method within a disease management A-CPS (Fig. 2) to identify disease hotspots. It proposes a novel method to represent the data of identified disease instances to capture their spatial relation in the farmland. Each diseased location and all its neighbors are analyzed to measure damage due to disease and identify hotspots of the disease in the farmland.

The rest of the article is structured as follows: Section II presents the problem statement with the proposed solution and its novelty while Section III discusses related prior research. Section IV introduces graphs and explores how graphs can be helpful in spatial analysis to propose a graph based method in Section V. The proposed method is verified experimentally in Section VI and Section VII concludes the article.
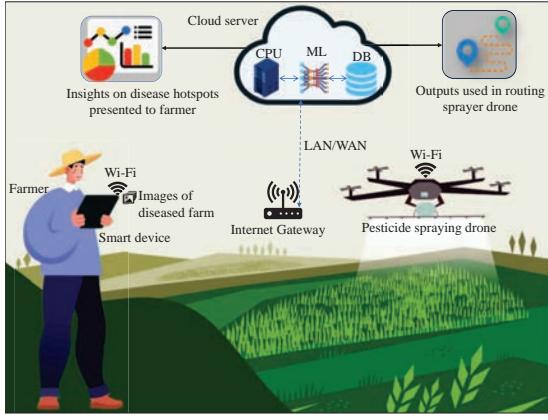
Figure 2: Overview of a disease management A-CPS.

II. NOVEL CONTRIBUTIONS OF THE CURRENT PAPER

A. The Problem Statement

Plant diseases can be caused by living organisms like bacteria, fungi, viruses while environmental factors like temperature, moisture, light conditions and polluted water playing their role. These diseases can be transmitted from an unhealthy plant to others by several natural carriers [8] like wind, soil, water, insects and direct contact. Besides these natural vectors, human beings act as spreaders of diseases directly by contacting diseased plants while scouting through fields or indirectly by using contaminated tools and equipment [9]. In the case of farming involving thousands of acres of land, characteristics of landscape across the farm also affect the spread of diseases [10]. Features like moisture retention, irrigation variation due to elevation, soil composition may not be the same throughout the farm and in parts, the environment would be favorable to diseases. Due to this heterogeneity, few parts of farmland may act as sources for diseases and play a crucial role in the spread of the disease. To best treat the disease, areas that act as sources (Hotspots) as represented in Fig. 3 have to be identified and efficiently treated.
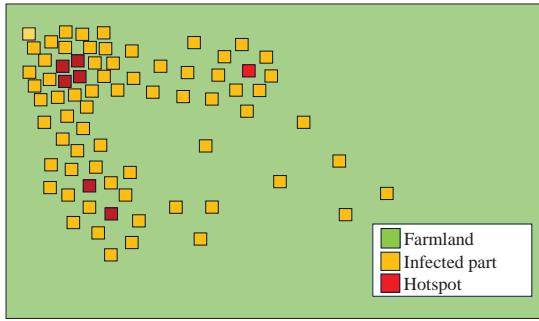

Figure 3: Representation of disease hotspots in farmland.

B. Proposed Solution of the Current Paper

Following the preceding discussion and interrelation between spatial patterns and the spread of diseases presented

in articles [11], [12], it can be inferred that the probability of a plant getting infected depends on its location and the health of plants closer to it. A hotspot can be defined as a specific area where the severity is higher than its surroundings. They have conditions that are highly favorable for pathogens and act as local sources of the disease. The current article "HIdentifier" presents graph graph-based method to analyze neighbors of each node and perform spatial analysis across farmland to identify hotspots of the detected disease.

C. Novelty and Significance of the Proposed Solution

The proposed method exhibits novelty of the followings:

1) Since the format/structure used to represent data plays a key role and can affect the results of the analysis performed, this article proposes an innovative representation suitable for spatial analysis to represent the data of identified diseased locations in a given farmland.
2) Identifies hotspots by calculating local incidence at each location and analyzing all its spatial neighbors.
3) The spatial analysis methods presented evaluate geometric distances between two diseased locations for effective modeling and accurate results.

III. RELATED PRIOR WORKS

This section discusses recent works in various domains related to the detection of hotspots. In [13], authors proposed different methods to detect hotspots in landscape ecology by generating probability density functions using kernel estimation and studying spatial auto-correlation using Moran's I and Getis-Ord G*. A frequency-based method that plots cumulative relative frequency table and identifies the points with sharp increase as hotspots was presented in [14]. The authors of articles [15] and [16] used DBSCAN to detect clusters with high density and identify them as hotspots. A ring-shaped hotspot detection was proposed in article [17] which draws ring-shaped boundaries where the intensity/concentration inside the ring is higher than the area outside it.

The above-discussed models are well suited for instances where each instance is uniquely identified, and they detect high-intensity clusters. To effectively manage disease spread, plants must be treated relatively. So, individual hotspot index has to be computed instead of detecting clusters. Though Moran's I and Getis-Ord G* can identify individual instances instead of clusters, the outputs are dependent on the feature value of each instance. When object detection algorithms are used to detect diseased plants, the output will be a bounding box around each diseased instance and the area of the box would be the same for most of the plants unless there is some overlap. In such cases, spatial auto-correlation methods would fail to detect the hotspots. So, this article proposes a novel method that represents data of diseased locations as a graph and properties of diseased locations as node features. Message passing is then performed on the graph to compute hotspot index of each diseased crop instance and identify hotspots. A summary of the research works has been presented in Table I.

Table I: A brief summary of relevant literature.

Research Works	Year	Methods	Remarks
Nelson et al. [13]	2008	Kernel estimation and spatial auto-correlation	Does not work for segmented image data
Bartolino et al. [14]	2011	Tangents drawn to cumulative relative frequency curve	All data points above threshold are considered hotspots
Nandana et al. [15]	2019	DBSCAN	High intensity clusters are identified instead of individual points
Nisa et al. [16]	2014	DBSCAN	High intensity clusters are identified instead of individual points
Eftelioglu et al. [17]	2016	Spatial scan and simulation using Monte Carlo	Intensity is high around plant; cannot be applied to diseases
HIdentifier	**2023**	**Message passing in graph**	**Works for all types of data and classifies data points**

IV. GRAPHS FOR SPATIAL ANALYSIS

A graph is a data structure primarily used to represent the relation between a collection of elements. Any graph [18] $G = (V, E)$ comprises of Nodes: $V = \{v_1, v_2, v_3, \ldots, v_n\}$ and Edges: $E = \{(v_i, v_j), (v_j, v_k), \ldots\}$ where nodes represent elements and edges represent the relation between the nodes it connects as depicted in Figure 4. Weighted Graphs have weights associated to each edge ($E = \{(v_i, v_j, w), (v_j, v_k, w'), \ldots\}$) and Unweighted Graph do not have any weights associated to edges. In a graph, all nodes that are directly connected to a node $u \in V$ by an edge form neighborhood of the node $N(u)$.
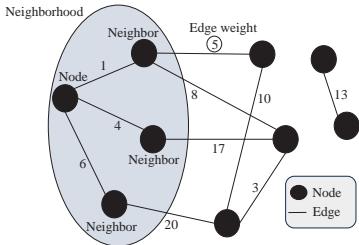

Figure 4: Example of a Graph.

The relationship between nodes in graphs is instrumental in learning interactions between nodes but nodes alone are not sufficient in representing information or context. Given that the spatial analysis tries to explore the relationship and connectivity between elements nearby, representing the data in the form of a graph $G = (V, E, F)$ with spatial attributes, characteristics as node features: $F = \{f_1, f_2, f_3, \ldots, f_n\}$ can unlock multiple methods to perform spatial analysis. Traditional Neural Network models can work on data with a feature set or images but not on graph data. So, new a model named Graph Neural Network was developed to work on graph data [19]. These models make use of Message Passing to update the feature vector/representation of each node to reflect the influence of its neighbors and perform ML tasks on updated feature vectors. Message passing [20] is an aggregation method involving the propagation of information between nodes to learn about its neighbors as in Figure 5.

The neighbor features thus learned are aggregated with current features of the node u to update its current feature vector h_u^k with its new feature vector h_u^{k+1} as per Eqn. 1:

$$h_u^{(k+1)} = \text{UPDATE}^{(k)}\left(h_u^{(k)}, \text{AGG}^{(k)}\left(\{h_v^{(k)} : v \in N(u)\}\right)\right), \quad (1)$$

where $h_u^{(k)}$ represents the feature vector of the node u in current k^{th} iteration and $h_u^{(k+1)}$ represents the feature vector of the
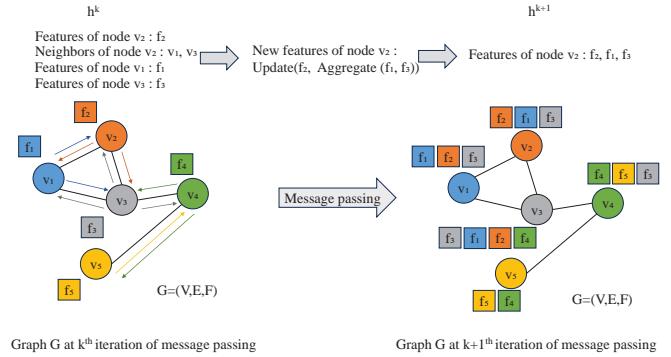

Figure 5: Illustration of message passing.

node u in the next $(k+1)^{th}$ iteration. The function $AGG^{(k)}$ is the aggregate function that aggregates feature vectors $h_v^{(k)}$ for every node v in the neighborhood $N(u)$ of node u. $UPDATE^{(k)}$ function computes the feature vector $h_u^{(k+1)}$ of node v for $(k+1)^{th}$ iteration by taking current feature vector $h_u^{(k)}$ and aggregated features computed by $AGG^{(k)}$ function as inputs.

Message passing helps in learning immediate neighbors and the structure of the graph. Whereas performing message passing repeatedly can diffuse features/information throughout the network/graph and unravel complex relations. Because of the expressive power of graphs and message passing, plant disease location data is represented as a graph for performing spatial analysis.

V. PROPOSED METHOD

This section presents a novel graph based spatial analysis method as illustrated by Figure 6 to identify hotspots and measure the severity of the disease in the farmland. The process of identifying or detecting a disease is not part of the scope of the article, it starts by loading an image of farmland with diseased locations segmented. The loaded image data flows through various processes as detailed below.

A. Graph Representation

A node in a graph represents an entity and edges represent their relationship. Since the proposed method is aimed to perform spatial analysis on locations of infected/diseased areas, each location is supposed to be represented as a node. A graph structure G is initialized and the image is processed to detect all the segments of the identified disease. For every segment detected, a node u and the area of the segment as a node feature f_1 represented by vector h_u is added to
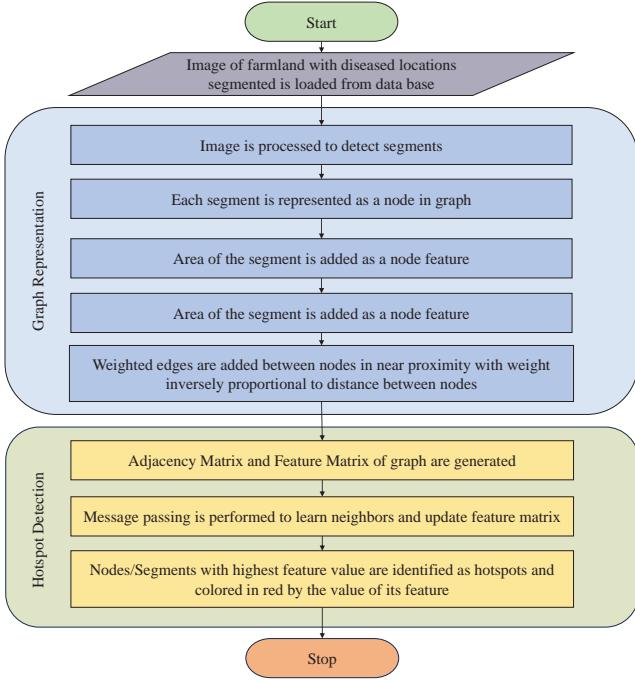
Figure 6: Flow chart of the proposed method.

the graph. If a plant is severely diseased or if plants have been closely planted, most of the plants right next to the diseased plant will also get infected and image segmentation can identify all of them as one large instance. So, adding the area of the segment as a node feature can embed the severity and presence of multiple diseased plants at a given location into the graph. Considering the mobility of insects, pathogens, water in a farm, disease from an infected plant will not spread beyond a few meters. Also, devices capturing images will have to be closer to the ground so that it can capture the visual features of plants needed to detect and classify the disease. Because of the above-mentioned reasons, we consider segments within a distance less than 10% of the image's width as neighbors. A weighted edge (v_i, v_j, w) with weight inversely proportional to the distance between neighboring i^{th} and j^{th} nodes is added between the pair. A self-edge (v_i, v_i) connecting a node to itself is also added for every node to aid message passing and the reason for doing this will be explained in subsequent processes. In consequence, the image is represented as a graph G with nodes $V = \{v_1, v_2, v_3, \ldots, v_n\}$, edges $E = \{(v_i, v_j, w), (v_j, v_k, w'), \ldots\}$, features $F = \{f_1\}$ similar to the one in Figure 5.

B. Hotspot Detection

In the case of plants, disease originates from a source plant and spreads to its surroundings. With time, the source plant will emerge as a center for a cluster of diseased plants where the density of diseased plants decreases with distance from the source plant [21]. Thus, a hotspot can be identified as a diseased location at the center of a high-intensity cluster. Given that every diseased location is represented as a node and edge

is present between locations that are close to each other, To find a hotspot, at least 2 hop neighbors of each location are to be analyzed. To accurately estimate the hotspots, in addition to considering the neighbors, the area of each neighbor location embedded as a node feature also must be considered. So, we compute the sum of areas of the current node and all its neighbors to replace its current feature value with a newly computed value. As a result, the updated feature of every node will represent the combined area of the node and its immediate neighbors. Repeating the same action will update the node feature with the area of 2 hop neighbors for the node and the feature value can be considered as a relative measure for the possibility of a node being a hotspot.

In scenarios where nodes are closely clustered, the above method produces the same feature value for multiple nodes in the cluster. As chances of infecting are higher in plants closer to the diseased plant factoring in the distance between neighbors can eliminate the risk of smoothening. This was the intuition behind adding edge weights inversely proportional to the distance between the nodes. Instead of just adding feature values of neighbors, neighbor feature values are multiplied by the weight of the edge between the node and neighbor to account for the distance between them.

Now that the solution is defined, we explore methods to implement it. To update the current feature of a node we are supposed to identify all its neighbors, multiply their feature values with corresponding edge weights and add sum of all the products to the current feature value as in Eqn. 2:

$$h_u^{(k+1)} = \sum \left(h_u^{(k)}, \sum \left(\{ h_v^{(k)} * W(uv) : v \in N(u) \} \right) \right). \quad (2)$$

The AGG function in Equation 1 is a summation function ($\sum$) and the UPDATE function is summation ($\sum$) of products of feature value $h_v^{(k)}$ and edge value $W(uv)$ of neighbors $N(u)$, it translates to Eqn. 2. This suggests that performing message passing twice on the generated graph G helps in identifying hotspots. To know the neighbors of a node for message passing, the Adjacency Matrix A of the graph G is generated. An adjacency matrix is a matrix of size $m \times m$ where m is the number of nodes in the graph. An entry $A[i][j]$ represents the edge weight if an edge exists between nodes i and j and 0 if there is no edge between them. $A[i]$ represents all the edges from a node i and if j is a neighbor of i, $A[i][j]$ denotes weight w of th edge (v_i, v_j, w) . The nodes of graph G have only one feature f_1 and feature representation h_u is just the feature value. Thus, matrix H representing feature vectors of all nodes is a matrix of size $m \times 1$ where $H[i]$ is the value of feature f_1 for i^{th} node. Performing matrix multiplication $A \times H$ will generate a column vector where i^{th} row is the sum of products of feature values and corresponding edge weights of all neighbors of ith node in graph, which is the aggregation function in Eqn. 2. So, Eqn. 2 can be rewritten as:

$$H^{(k+1)} = H^{(k)} + \left(A \times H^{(k)} \right). \quad (3)$$

Adding a self-connecting edge (v_i, v_i) will generate an adjacency matrix that considers a node as a neighbor of itself and

adds its feature value to the sum of the feature of its neighbors in message passing. So, there is no need to explicitly add $H^{(k)}$ and $A \times H^{(k)}$. This was the reason for adding a self-edge while creating the graph G. Finally, the message passing Eqn. 3 can be simplified to Eqn. 4, performing this on the generated graph will update feature values as expected and the ones with higher feature value will be considered as hotspots:

$$H^{(k+1)} = A \times H^{(k)}. \qquad (4)$$

VI. EXPERIMENTAL RESULTS AND DISCUSSION

The graph-based solution presented in Section V has been implemented using Python and a Python library called NetworkX for the creation and manipulation of graphs. For experimental verification, a data set [22] which contains annotated images of diseased apple leaves has been used. But in practice segment images of diseased farmland will be used. An image of a leaf with its annotations is loaded and bounding boxes are drawn around the infected part as in Fig. 7.
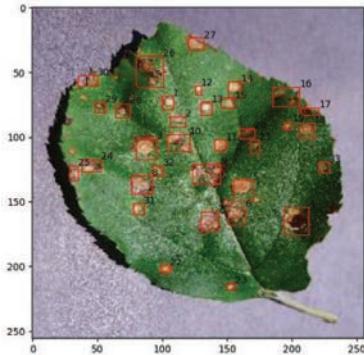
Figure 7: Input image with bounding boxes.

This image is now processed to generate a graph shown in Figure 8 with node features and weighted edges as specified in Section V-A. The graph thus generated had 33 nodes out of which 26th node had the highest feature values so was colored in dark red.
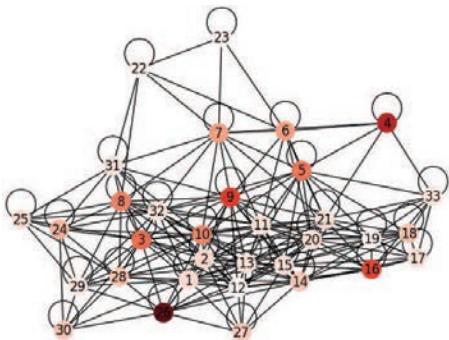
Figure 8: Graph representation of input image.

Local Getis-Ord G^* compares the average feature value of the cluster around a data point with the global average to output a high value if the local average is greater than the global average and vice versa. Local Moran's I compares features of the data point with the rest in the cluster to determine their spatial auto-correlation. The value is $+1$ if the feature value is high and surrounded by high feature values. Thus, a hotspot is supposed to have high values for Local Getis-Ord G^* and Local Moran's I. CNN methods can only detect diseases and segment them but cannot know the intensity of the disease. So, the area of the bounding measure serves as a measure of intensity at the location. Though the nodes 10, 11, 13 are at the center of all detected disease spots Local Getis-Ord G^*, Local Moran's I values in Figure 9 failed to identify them as hotspots.

Figure 9: Local Moran's I and Local Getis-Ord G^* values for nodes of graph in Figure 8.

Figure 10 represents the feature matrix and adjacency matrix of the graph in Figure 8. These matrices will be multiplied to perform the first iteration of message passing. The resultant matrix is again multiplied with the adjacency matrix to complete the second iteration of message passing.

Figure 10: Feature matrix and Adjacency matrix of the graph.

Figure 11 represents the graph after two rounds of message passing. Since, nodes 2, 10, 11, 13 were at the center with a greater number of neighbors, after message passing, they had the highest feature value and can be considered hotspots. Accordingly, diseased areas of the leaf were colored relative to their probability of being a hotspot in Figure 12.

A similar trend can be observed in Local Moran's I and Getis-Ord G^* values in Figure 13 computed from nodes after Message Passing.

Thus, the presented HIdentifier can efficiently perform spatial analysis using message passing to understand relationships with neighbors and detect disease hotspots in agriculture. It was able to identify individual instances rather than clusters to assist downstream tasks like targeted pesticide spraying and relative disease control.
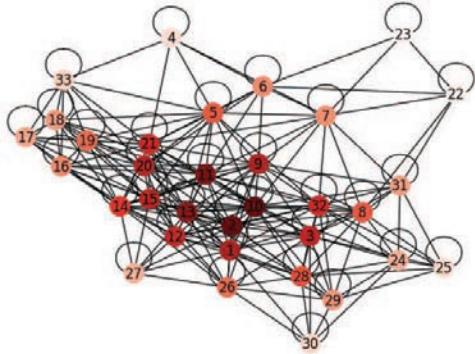
Figure 11: Graph representation after Message Passing.
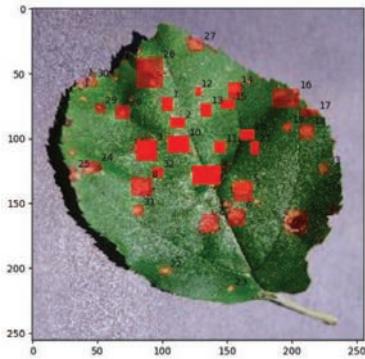
Figure 12: Updated image of leaf with hot spots identified.

VII. CONCLUSION

This article presented an efficient method to perform spatial analysis in disease-affected farmland to identify hotspots of the disease. It proposed a novel method to represent the data to aid in spatial analysis. This solution outputs insights that can help farmers understand disease dynamics for efficient disease management through smart agriculture solutions. But the proposed solution cannot predict areas that can be affected consequently. With help of IoAT and various sensors, environmental conditions and information related to various parts of the farm can be acquired. Incorporating these as features of each node and using recent methods like GNN and GCN to proactively predict the future spread of the disease can be explored as future works.

Figure 13: Local Moran's I and Local Getis-Ord G* values after Message Passing.

REFERENCES

[1] R. M. Pelczar, M. C. Shurtleff, A. Kelman, and M. J. Pelczar, "Plant disease," *Encyclopedia Britannica*, 2023, https://www.britannica.com/science/plant-disease, Accessed 5 June 2023.

[2] R. N. Strange and P. R. Scott, "Plant disease: a threat to global food security," *Annu. Rev. Phytopathol.*, vol. 43, pp. 83–116, 2005.

[3] D. M. Rizzo, M. Lichtveld, J. A. Mazet, E. Togami, and S. A. Miller, "Plant health and its effects on food safety and security in a one health framework: Four case studies," *One health outlook*, vol. 3, pp. 1–9, 2021.

[4] A. Mitra, S. P. Mohanty, and E. Kougianos, "Smart Agriculture – Demystified," in *Internet of Things - Advances in Information and Communication Technology*. Springer Nature Switzerland, 2024, pp. 405–411.

[5] P. K. Tripathy, A. K. Tripathy, A. Agarwal, and S. P. Mohanty, "My-Green: An IoT-Enabled Smart Greenhouse for Sustainable Agriculture," *IEEE Consumer Electronics Magazine*, vol. 10, no. 4, pp. 57–62, July 2021.

[6] I. Buja, E. Sabella, A. G. Monteduro, M. S. Chiriacò, L. De Bellis, A. Luvisi, and G. Maruccio, "Advances in plant disease detection and monitoring: From traditional assays to in-field diagnostics," *Sensors*, vol. 21, no. 6, pp. 21–29, 2021.

[7] M. T. Kuska, R. H. Heim, I. Geedicke, K. M. Gold, A. Brugger, and S. Paulus, "Digital plant pathology: a foundation and guide to modern agriculture," *Journal of Plant Diseases and Protection*, vol. 129, no. 3, pp. 457–468, 2022.

[8] S. Singh, L. Awasthi, and A. Jangre, "Transmission of plant viruses in fields through various vectors," in *Applied Plant Virology*. Elsevier, 2020, pp. 313–334.

[9] B. Ranawaka, S. Hayashi, P. M. Waterhouse, and F. F. de Felippes, "Homo sapiens: The superspreader of plant viral diseases," *Viruses*, vol. 12, no. 12, pp. 14–62, 2020.

[10] M. Plantegenest, C. Le May, and F. Fabre, "Landscape epidemiology of plant diseases," *Journal of the Royal Society Interface*, vol. 4, no. 16, pp. 963–972, 2007.

[11] L. A. Real and P. McElhany, "Spatial pattern and process in plant–pathogen interactions," *Ecology*, vol. 77, no. 4, pp. 1011–1025, 1996.

[12] J. B. Ristaino and M. L. Gumpertz, "New frontiers in the study of dispersal and spatial analysis of epidemics caused by species in the genus phytophthora," *Annual Review of Phytopathology*, vol. 38, no. 1, pp. 541–576, 2000.

[13] T. A. Nelson and B. Boots, "Detecting spatial hot spots in landscape ecology," *Ecography*, vol. 31, no. 5, pp. 556–566, 2008.

[14] V. Bartolino, L. Maiorano, and F. Colloca, "A frequency distribution approach to hotspot identification," *Population Ecology*, vol. 53, pp. 351–359, 2011.

[15] G. Nandana, S. Mala, and A. Rawat, "Hotspot detection of dengue fever outbreaks using dbscan algorithm," in *Proc. 2019 9th International Conference on Cloud Computing, Data Science and Engineering (Confluence)*, 2019, pp. 158–161.

[16] K. K. Nisa, H. A. Andrianto, and R. Mardhiyyah, "Hotspot clustering using dbscan algorithm and shiny web framework," in *Proc.2014 International Conference on Advanced Computer Science and Information System*, 2014, pp. 129–132.

[17] E. Eftelioglu, S. Shekhar, J. M. Kang, and C. C. Farah, "Ring-shaped hotspot detection," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 12, pp. 3367–3381, 2016.

[18] M. Newman, *Networks*, 2nd ed. Oxford university press, Sep. 2018, pages 1-1.

[19] R. Merrit, "What are graph neural networks?" Oct. 2022, last accessed June 5 2023. [Online]. Available: https://blogs.nvidia.com/blog/2022/10/24/what-are-graph-neural-networks/

[20] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl, "Neural message passing for quantum chemistry," *arXiv*, vol. 1704.01212, 2017.

[21] P. A. Arneson, "Plant disease epidemiology," *The Plant Health Instructor*, 2001.

[22] A. Mitra, "Annotated apple leaf disease dataset - mask rcnn," 2023, accessed on date 6/5/2023. [Online]. Available: https://www.kaggle.com/datasets/amitra19/annotated-apple-leaf-disease-dataset-mask-rcnn