An IoT Environmental Data Collection System for Fungal Detection in Crop Fields

Thomas Truong, Anh Dinh, Khan Wahid
Department of Electrical and
Computer Engineering
University of Saskatchewan
Saskatoon, Canada
Email: thomas.truong@usask.ca

***Abstract*—There is a need for a system which provides real-time local environmental data in rural crop fields for the detection and management of fungal diseases. This paper presents the design of an Internet of Things (IoT) system consisting of a device capable of sending real-time environmental data to cloud storage and a machine learning algorithm to predict environmental conditions for fungal detection and prevention. The stored environmental data on conditions such as air temperature, relative air humidity, wind speed, and rain fall is accessed and processed by a remote computer for analysis and management purposes. A machine learning algorithm using Support Vector Machine regression (SVMr) was developed to process the raw data and predict short-term (day-to-day) air temperature, relative air humidity, and wind speed values to assist in predicting the presence and spread of harmful fungal diseases through the local crop field. Together, the environmental data and environmental predictions made easily accessible by this IoT system will ultimately assist crop field managers by facilitating better management and prevention of fungal disease spread.**

Keywords*— *Internet of Things, Environmental data, Weather prediction, Support vector machines

I. Introduction

Ensuring high crop production yield is critical in maintaining global food security [1][2][3]. Global crop demand is expected to double by 2050 [4], requiring crop yield increases of 2.4% annually [3]; however, current global efforts are only increasing crop yield by about 1.3% annually [2][3]. A large portion of this deficit is because 30% of the land used for crop growing has stagnant or falling annual crop yields due to variable and unfavourable environmental conditions such as the presence of harmful fungal diseases [5]. There are two prominent fungal spores which lead to disease: *Fusarium spp.* (responsible for fusarium head blight) and *Puccinia graminis* (responsible for wheat rust). Fusarium head blight and wheat rust are two devastating plant diseases among several other pathogen species which have a large economic impact on crop production worldwide [6]. The growth and spread of these spores are heavily influenced by environmental conditions.

One method to increase crop yield is to proactively protect crops by predicting future environmental conditions which promote the spread of fungal diseses. Current weather predictions readily available to the urban public are not always provided for rural areas where crops are generally grown, resulting in inaccurate predictions for field managers to analyze. Thus arises the need for a system which measures and predicts real time environmental conditions for rural fields.

This paper presents the hardware and software design of an IoT environmental data collection and environmental condition prediction system. In particular, this paper focuses on how the collected data are processed to predict air temperature and relative air humidity, two environmental factors that have significant correlation with the presence of *Fusarium spp.* and *Puccinia graminis* in the field [7]. Additionally, wind speeds are also predicted in order to assist in predicting the ability of the fungal diseases to spread via airborne spores.

Ultimately, this system helps inform crop field managers of current and future environmental conditions to prevent loss in yield. Moreover, this IoT system allows field managers and researchers to remotely retrieve environmental data from rural areas without needing to physically retrieve a storage device from the field.

II. Hardware Design

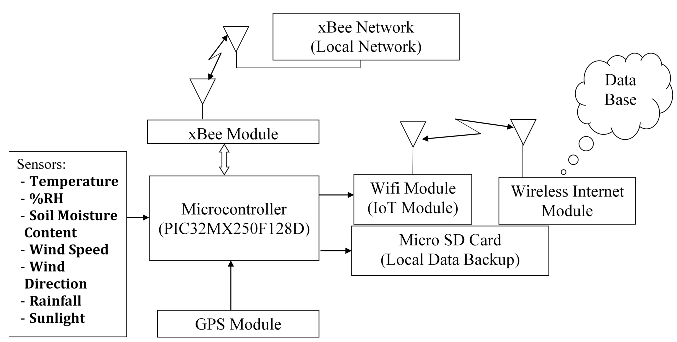

Fig. 1. The high level hardware block diagram for a single environmental data collection device.

Fig. 1 shows the high level hardware block diagram for a single environmental data collection device. The device contains the hardware responsible for data collection in the field. All hardware is powered by a solar panel charger and a 12V lead-acid battery. The battery autonomy ranges from 3 days up to 7 days depending on the sleep duration of the device. The description of the critical blocks for the IoT implementation are as follows:

978-1-5090-5538-8/17/$31.00 ©2017 IEEE

A. Sensors

Currently the device contains components to measure air temperature, relative air humidity (%RH), soil moisture content, wind speed, wind direction, rain fall, and sunlight intensity. The sensors are sampled by the microcontroller at a user specified interval. Up to 13 analog sensor devices including sensors such as a fungal spore detector can easily be added in the future.

B. Wireless Internet Module

The Wireless Internet Module is an LTE HUA8372 WiFi dongle distributed by Bell. The module provides WiFi at any location with a cellular network.

C. WiFi Module

The WiFi module is an imp001 manufactured by Electric Imp. The imp001 is a low-profile device designed to streamline the process of developing an IoT system. The imp 001 receives sensor data from the microcontroller via the universal asynchronous receiver/transmitter (UART) pins and then relays the sensor data to cloud storage using the internet provided by the Wireless Internet Module. The WiFi Module in conjunction with the Wireless Internet Module provides the IoT functionality for the hardware portion of the system.

III. Internet of Things Software Design

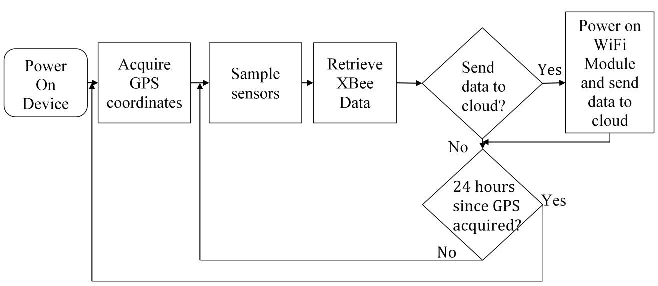

Fig. 2. The high level software block diagram for the microcontroller.

Fig. 2 shows the high level software block diagram for a single environmental data collection device. The microcontroller is programmed in C using MPLAB XC32, an integrated development environment built by Microchip Technology. The microcontroller is programmed to sample and push data to the cloud at user specified intervals. GPS Coordinates are acquired on power resets and at 24 hour period intervals. The microcontroller has the ability to power off unused modules and to conserve power in the field.

The WiFi Module is programmed in Squirrel, an object-oriented programming language. When the WiFi Module is powered on, the Wifi Module receives data through the serial pin from the microcontroller. The WiFi Module then relays the data to Sparkfun Data, a free online cloud storage service provided by Sparkfun. Additionally, the GPS coordinates are also relayed onto a website built with basic HyperText Markup Language (HTML) and Javascript. The website uses Google Maps to display the location of the devices in real-time.

IV. Support Vector Machine Regression Software Design

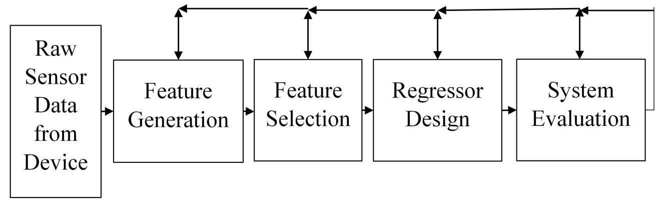

Fig. 3. The high level software block diagram for the short-term (day-to-day) environmental condition prediction system.

Fig. 3 shows the high level software block diagram for the short-term environmental condition prediction system. A machine learning approach to environmental condition prediction is used as it only requires the readily available sensor data in cloud storage. Additionally, machine learning approaches have been fairly successful in predicting short-term environmental conditions. SVMr based wind speed predicting systems have been designed for wind turbine power production [8] and solar irradiance predicting systems have been designed for photovoltaic power production [9][10]. In both cases it was found that the machine learning systems used to predict wind speeds and solar irradiance were reliable enough for their application in power production models. For applications in agriculture, machine learning has been used to predict daily evapotranspiration [11] as well as daily atmospheric temperature [12]. SVMr has been selected as the regressor of choice for this project for its aforementioned success in weather prediction. Additionally, Ridge Regression and an ensemble regressor method, Random Forests, were also implemented. Ultimately, SVMr performed optimally in terms of accuracy and computational complexity.

Air temperature, relative air humidity, and wind speeds were the focus of the SVMr optimization for their high correlation to *Fusarium* presence [7]. In the future, more environmental conditions can be predicted with slight modification to the following proposed design.

After retrieving the data from the online storage from a remote computer in a lab, Python 3.0 is used to create a feature vector of the form:

$$\mathbf{x} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1M} \\ x_{21} & x_{22} & \cdots & x_{2M} \\ \vdots & \vdots & \ddots & \vdots \\ x_{(L-1)1} & x_{(L-1)2} & \cdots & x_{(L-1)M} \end{bmatrix} \tag{1}$$

Each of the L rows directly corresponds to each of the days collected, ie. row 4 corresponds to the data collected on day 4. Each of the M columns directly corresponds to a raw value for a specific sensor at a specific time for that day, ie. column 3 will correspond to the air temperature sensor measurement at 00:00. Note that although there may be portions data collected for day L, row L is omitted until all M elements are measured for row L.

One truth vector

$$\mathbf{Y_T} = \begin{bmatrix} Y_{T2} & Y_{T3} & \dots & Y_{T(L-1)} \end{bmatrix} \tag{2}$$

is created for each of the air temperature, relative air humidity, and wind speed measurements. The three vectors have elements containing the daily average of the measured values for air temperature, relative air humidity, and wind speed. The values are calculated based on the elements in $\mathbf{x}$. Note that Y_{T2} is the first element in $\mathbf{Y_T}$ and is generated by calculating the average daily value for the corresponding measurement on day 2; however, Y_{T2} is treated by the machine learning algorithm as the truth value for measurements made on day 1. For example, the first element in the truth vector $\mathbf{Y_T}$ for the air temperature contains the average daily air temperature for day 2. This structure allows for training machine learning algorithms to be trained predict future values based on past and present data.

Then, using the feature vector $\mathbf{x}$ and the truth vector $\mathbf{Y_T}$, an SVMr machine learning algorithm using a radial basis function kernel provided by the Scikit-learn library is trained. From here, a prediction for day L is made and is placed in a prediction vector of the form:

$$\mathbf{Y_P} = \begin{bmatrix} Y_{PL} \end{bmatrix} \tag{3}$$

Once a complete set of sensor data for day L is collected, $\mathbf{x}$ is appended with row L containing M elements and $\mathbf{Y_T}$ is appended with column L containing the value Y_{TL}. The performance of the system is evaluated through calculation of an absolute error.

$$AbsoluteError = |Y_{TL} - Y_{PL}| \tag{4}$$

Then the SVMr algorithm is retrained with newly updated vectors $\mathbf{x}$ and $\mathbf{Y_T}$ and then new predictions are made for day L+1 so that $\mathbf{Y_P}$ is appended to become:

$$\mathbf{Y_P} = \begin{bmatrix} Y_{PL} & Y_{P(L+1)} \end{bmatrix} \tag{5}$$

This process of iteratively updating vectors $\mathbf{x}$ and $\mathbf{Y_T}$ and then retraining the SVMr algorithm to predict future average air temperature, relative air humidity, and wind speed values placed in $\mathbf{Y_P}$ each day forms a short-term weather prediction system.

Furthermore, simple optimization algorithms are performed at each retraining iteration to minimize absolute error. The first optimization is a reduction of complexity in $\mathbf{x}$. This is done by creating a new feature vector $\mathbf{x_{opt}}$ containing a percentage of the most statistically significant columns in $\mathbf{x}$ based on the results of univariate statistical tests [13]. The optimized feature vector $\mathbf{x_{opt}}$ is then used in place of $\mathbf{x}$ in training the SVMr algorithm. The second optimization involves a grid search of parameters C (penalty parameter of the error term) and ϵ ("slack" parameter associated with the training loss function in SVMr) in the SVMr algorithm for optimal values [13]. A more detailed formulation of these parameters and the SVMr algorithm can be found in the documentation of Scikit-Learn library [13].

V. Results

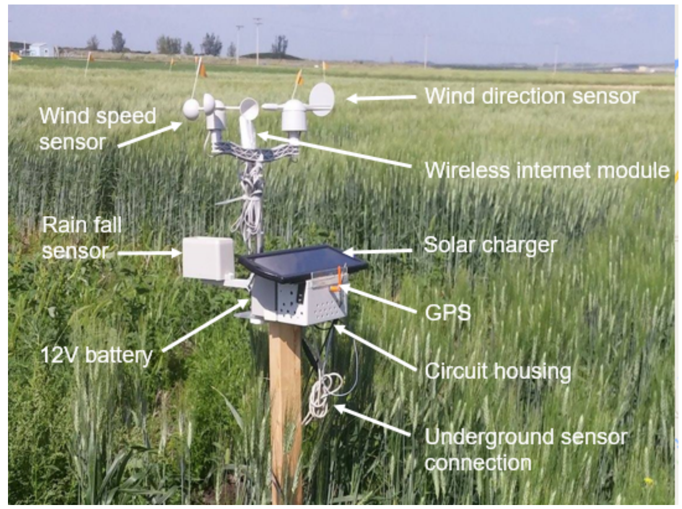

Fig. 4. The environmental data collection device in the field.

Fig. 4 shows the operation of a single environmental data collection device in the field. The device sends its geographical location to a website containing a Google Maps JavaScript application program interface (API). The API takes the geographical information of the device and places a marker on the Google Map at the device location. The marker contains a hyperlink which leads to the Sparkfun database where the device measurements are retrieved.

A total of 3 devices were set-up around Saskatoon, Saskatchewan between the months of July, August, and September in 2016. The most successful device collected data from July 30th, 2016 to September 9th, 2016. The device required no maintenance during this time and was only taken down once the crops were harvested by the field managers. Data were collected from the sensors at an interval of 1 sample every 10 minutes. The machine learning algorithm outlined in section IV was developed and trained using this device's data.

Fig. 5 shows the predicted average air temperature values for each day with the SVMr algorithm predictions having an average absolute error of 2.02^0C. Fig. 6 shows the predicted average relative air humidity values for each day with an average absolute error of 6.98%. Finally, Fig. 7 shows the predicted average wind speed for each day with an average error of $1.87km/h$. These predictions are sufficient so that crop field managers can reliably determine future high temperature and relative humidity days, conditions which are highly correlated with the presence of harmful fungal diseases [7]. Additionally, the wind speeds will help determine the ability of the fungal diseases to spread via airborne spores.

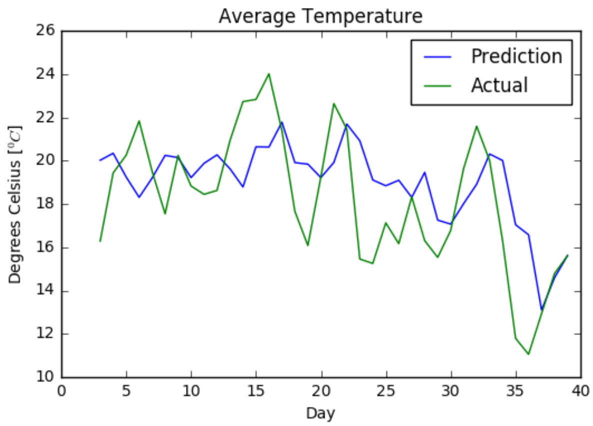

Fig. 5. Average air temperature predictions compared with the actual measured values for each day.

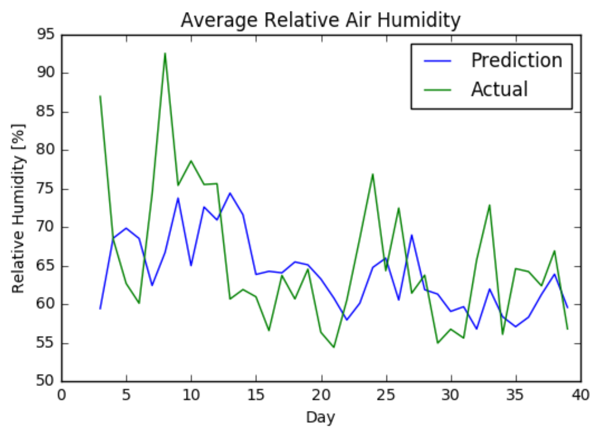

Fig. 6. Average relative humidity predictions compared with the actual measured values for each day.

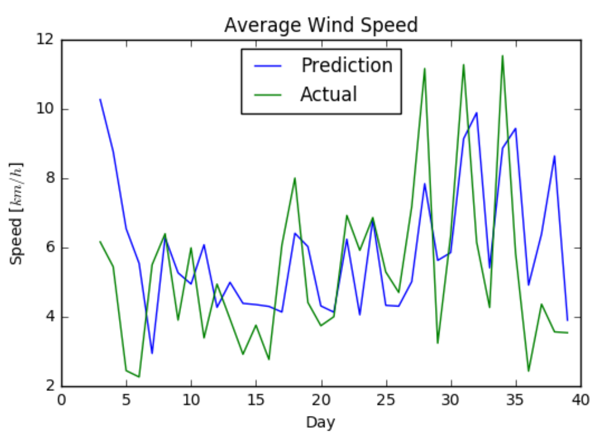

Fig. 7. Average wind speed predictions compared with the actual measured values for each day.

VI. Conclusions

This paper presented the design of an IoT based system which provides easily accessible real-time local environmental data in rural crop fields. The devices were set-up and collected data at 10 minute intervals between the months of July, August, and September in 2016 around various crop fields in Saskatoon, Saskatchewan. The data are pushed in real-time to easily accessible cloud storage, providing researchers and crop field managers with accurate environmental data without the need for visits to the crop field to retrieve local data. A basic machine learning algorithm based on SVMr was designed to predict average air temperature, relative air humidity, and wind speed values. On average, the SVMr based machine learning algorithm was able to predict daily average air temperature, relative air humidity, and wind speed values to within 2^0C, 7%, and $2km/h$ true value. The accuracy of these predictions is expected to improve as more experimental data is collected by these devices in the future.

Ultimately, this system will be used to improve the detection of fungal diseases and predict how the diseases will spread in the crop fields.

Acknowledgment

The authors would like to acknowledge the support from the Agriculture Development Fund, Saskatchewan, Canada.

References

[1] N. Fahlgren, M. Gehan, and I. Baxter, "Lights, camera, action: high-throughput plant phenotyping is ready for a close-up,," *Current Opinion in Plant Biology*, 2015.

[2] L. Li, Q. Zhang, and D. Huang, "A review of imaging techniques for plant phenotyping," *Sensors*, 2014.

[3] J. Araus and E. Cairns, "Field high-throughput phenotyping: the new crop breeding frontier," *Trends in Plant Science*, 2014.

[4] D. Tilman, C. Balzer, J. Hill, and B. L. Befort, "Global food demand and the sustainable intesification of agriculture," in *Proceedings of the National Academy of Sciences of the United States of America*.

[5] D. Ray, N. Ramankutty, N. Mueller, P. West, and J. Foley, "Recent patterns of crop yield growth and stagnation," *Nature Communications*, 2012.

[6] V. Panwar, B. McCallum, and G. Bakkeren, "Host-induced gene silencing of wheat leaf rust fungus puccinia triticina pathogenicity genes mediated by the barley stripe mosaic virus," *Plant Molecular Biology*, 2013.

[7] A. Schaafsma and D. Hooker, "Climatic models to predict occurrence of fusarium toxins inwheat and maize," *International Journal of Food Microbiology*, 2007.

[8] S. Salcedo-Sanz, E. G. Ortiz-Garc, ngel M. Prez-Bellido, A. Portilla-Figueras, and L. Prieto, "Short term wind speed prediction based on evolutionary support vector regression algorithms," *Expert Systems with Applications*, 2011.

[9] F. Wang, Z. Zhen, Z. Mi, H. Sun, S. S. d, and G. Yang, "Solar irradiance feature extraction and support vector machines based weather status pattern recognition model for short-term photovoltaic power forecasting," *Energy and Buildings*, 2015.

[10] J. Li, J. K. Ward, J. Tong, L. Collins, and G. Platt, "Machine learning for solar irradiance forecasting of photovoltaic system," *Renewable Energy*, 2016.

[11] A. F. Torres, W. R. Walker, and M. McKee, "Forecasting daily potential evapotranspiration using machine learning and limited climatic data," *Agricultural Water Management*, 2011.

[12] Y. Radhika and M. Shashi, "Atmospheric temperature prediction using support vector machines," *International Journal of Computer Theory and Engineering*, 2009.

[13] scikit-learn developers. (2016) Documentation scikit-learn: machine learning in python. Last accesed January 30, 2017. [Online]. Available: http://scikit-learn.org/stable/documentation.html