An Accessible Approach to Agricultural Monitoring and the Viability of machine learning and A.I. Methods to determining soil pH via Imaging

Maciej Krzysztof Mucha

Student ID: 2156739
Programme Name: BSc Computer Science
Supervisor: Carl Wilding
Word Count: 8587

April 2023

Abstract

Detrimental environmental issues we face in the modern day have a disproportionate effect on agricultural communities, especially those in the emerging and developing world. While top-down aid has been provided by governments and NGOs alike, and precision farming technologies such as IoT and AI have been widely used in the developed world, these specific projects have not directly addressed the problem of depleting resources and the lack of capital hindering affected communities from addressing these issues in a long-term and sustainable manner.

I believe that the technological sophistication used in the precision farming sector can be implemented in a low-cost fashion with minimal detriment to quality. To achieve this, I design a modular architecture for a low-cost agricultural monitoring system and implement AI methods to expand its capabilities and reduce cost. Furthermore, I posit that the research and approaches in the field of AI for soil pH analysis require improvement, and I set out to do this by introducing a new AI model which not only out-performs existing models but also takes a more holistic approach to the problem and is, therefore, more reliable

Proving this approach to be viable via a successful prototype while providing sound argumentation for its scalability and preserving it's simplicity, I believe I lay the foundation for more action to be taken in regard to implementing these solutions.

Acknowledgements

I would like to thank my supervisor Carl Wilding who provided me with helpful feedback and support throughout the entire course of my project.

I would also like to thank Dr Ian Styles for taking the time to help me confirm some of my suspicions that arose from the data analysis I conducted.

Finally, I want to thank Dr Paul Levy, who inspected my project and provided me with feedback and challenging questions which helped me to develop the rigour with which I justify my approaches to the problem I chose to confront.

Guide to the paper

In the literature review, I outline the background behind the specific issues I am addressing, which provide the motivation for this project. I discuss the consequences and feedback loops pertaining not just to the environment but also to their socio-economic implications to highlight the necessity for the proposed action. Understanding the background issues in terms of their geographical and socio-economic components provides the foundation for the reasoning I invoke to justify the specific methodology in confronting them.

In the methodology chapter, I propose the specifics of how to confront the previously outlined issues such that I account for the successes and shortcomings of past approaches. This includes the basic hardware, software, and AI as a subsection of the software that will comprise the project.

The results and implications chapter shows the results of the experiments and designs undertaken during the course of the project. If the result was unexpected or necessitated more: experimentation, analysis, or testing, this is mentioned as an implication of the result and elaborated on further. The discussion of the results is limited to those which directly caused the further investigation. These only pertain to the specifics of the methods and experimental results and do not directly aim to answer nor evaluate the initial questions or goals set out at the beginning of the project. This is reserved for the final chapter.

The Conclusions chapter takes a holistic approach to evaluate the success of the project as a whole. It links back to the initial goals of the project and outlines how it could be improved and continued if more time was available.

Finally, this project includes many figures as a result of thorough data analysis and experimentation. The figures necessary to illustrate a concept or find are included. However, I did not include all figures in this paper as many of them point to a similar conclusion or are not explicitly necessary to convey a certain point, were these figures to be included, the document would be ludicrously long. However, there are figures available for most results and may be found in their corresponding Jupyter notebooks. They are interactive and allow the user to navigate 3d plots and highlight any interesting points. The notebooks also demonstrate the step-by-step process and code to obtain these results. I encourage the reader to take a look at them nonetheless at their leisure.

Contents

Chapter 1

Literature Review

1.1 Motivation

1.1.1 Soceo-economic and Environmental background

A large portion of the world's environmental issues may be attributed to increased demand for natural resources due to a combination of population growth and growing purchasing power parity (PPP) among pre-existing populations. While increased populations lead to increased demand for resources rather trivially, increased PPP leads to increased demand for natural resources per individual (whether this demand is a necessary one or a luxury). Increased economic power leads to increased demand for goods and services, which in turn require natural resources. The magnitude of resources required for just one of these items is often underestimated, e.g. the average water footprint for a cotton clothing product is approximately 3233 cubic metres (3,233,000 litres) of water [1]. As water is arguably one of the most important resources and foundational to our necessary and surplus desires, I will continue using it as the primary example of our resource deficiencies.

The population has more than doubled since 1970 [2]. The current population sits at 7.98 billion people and is projected to increase by a further 18.6% by 2050 [3]. Naturally, this will only increase the already present significant demand for resources. Water demand has increased by 600% over the last 100 years [4] and is expected to increase by another 20%-30% by 2023 [5]. Water is not the only resource that will suffer, the Global Symposium on Soil Erosion (2019) organised by the Food and Agriculture Organization of the United Nations, found that 30% of the global land area has faced significant soil degradation, and this figure could increase to 90% by 2050. This is significant as 90% of global food comes directly from agricultural soil [6] and it suffices to say that food is essential to our survival as a species.

Climate Change further exacerbates the above-listed issues. Increasing sea levels causes saltwater encroachment on freshwater resources such as groundwater or other stores, further reducing stores of available water for consumption's [7]. The IPCC stated "At 2°C or higher global warming level in the mid-term, food security risks due to climate change will be more severe, leading to malnutrition and micro-nutrient deficiencies, concentrated in Sub-Saharan Africa, South Asia, Central and South America and Small Islands. Global warming will progressively weaken soil health and ecosystem services such as pollination, increase pressure from pests and diseases, and reduce marine animal biomass, undermining food production in many regions on land and in the ocean.". The consequences of weakened soil health and other ecosystem services decrease food security and pile on to the existing environmental problems discussed.

The issue of climate change cannot be understated with respect to the feedback loops it triggers. As an example, increased sea temperatures decrease the capacity for oceanic carbon sequestration, in turn causing increased natural emissions of greenhouse gasses which warm the atmosphere further and consequently creates a cycle of decreased oceanic sequestration potential.

Though this is only one example of dangerous feedback loops, are many others which exacerbate the present environmental issues, which ultimately translate to socioeconomic problems. Unfortunately, indicating that the issues discussed in this paper will not go away anytime soon, therefore alongside current attempts at mitigation, we must, as soon as possible, create adaptation strategies for a new environment with decreased essential resources.

1.1.2 Past and Possible Future Solutions

The combination of increasing demand for primary resources, coupled with decreased supply of the resources required to acquire them (and environmental issues increasing the rate of the supply-demand disparity), causes a significant challenge for the global population. The supply-demand curve for food is set to cause the price of essential resources to outgrow what many can afford, especially in the developing world.

Many schools of thought exist with regard to tackling the problem. We can try to increase supply through means such as desalination in tandem with regenerative agriculture. A notable case study of Israel shows this approach to be successful on the whole, 85% of water consumed in Israel comes from desalination plants [8]. However, these top-down schemes have a very high start-up cost. They could only realistically be afforded by large-scale organisations such as governments of developed nations or companies with a lot of capital to spare e.g. the startup cost of the desalination plant in Tel Aviv was approx $500 million [9]. Not to mention it is limited to non-landlocked nations, and those nations which are landlocked are at an economic disadvantage due to a lack of means to export goods. This means this strategy is unavailable to the nations that are most at risk of suffering from a shortage of essential resources.

Any solution which may aim to decrease demand in the first place is unlikely to succeed. While many communities in developed nations have attempted to do just that, the impact of this community action is minimal (in the global context) as seen by the fact that consumption in developed nations is at an all-time high [10]. Halting climate change and resource depletion to a degree that could have a global impact would likely require action in rapidly emerging countries such as India or China [10]. We must, however, take into account that these emissions come from industrial action necessary to maintain their growth. We can think of these countries being on the "take off" / "drive to maturity" stage on Rostow's model of development and certainly edging on the developed nation category if not already there.

We may loosely equate this stage to the industrial revolutions in the West during the 18th and 19th centuries, where similar levels of relative emissions and resource consumption occurred. This gave the West a huge economic advantage over the rest of the world. As a result, many political scientists and economists posit that it is unrealistic to expect the emerging world to sacrifice its economic development and its people's prosperity despite the understanding of its environmental impact.

If we have excess demand for natural resources, a deficit of supply, and we cannot realistically decrease demand we must look at ways to slow down the depletion of supplies and maximise the efficiency with which they are used (this, if done well, will lead to increased supplies in the long term). Having argued the importance of this area of research and development, I propose that the most promising solution to this global issue is the use of two main technologies, AI and IoT.

1.2 Technological approaches motivating the project

These two technologies have proven to be highly effective in combating these issues. Microsoft's Farm Beats project is the largest player in this specific field. They have taken the approach of using drones as well as AI to monitor agricultural plots, as well as cattle farms that need attention. This allows workers to direct their focus and resources solely on the areas that pertinently

require it. This saves not only time but the extra information that allows for more specialised and worthwhile action to benefit the agricultural growth of the area. Farm Beats is arguably the most innovative and effective project thus far, but it has some drawbacks. While Farm Beats aims to minimise prices for affordability, their projects have mainly been implemented in the USA, and the prices may be viable for US consumers, but for those with less income in the developing world, it is not. One of Farm Beats' most important (and arguably underrated) actions is using unused radio waves otherwise reserved for TV channels for sending long-range WiFi signals. This unlocks the potential for IoT and other technologies to be implemented in a great number of areas, including the developing and emerging world.

It has, however, been shown that it is possible to create technological solutions (albeit of a slightly different form) in the developing world. In many areas of Africa, soil moisture sensors connected over long-range internet connections hosted on unused radio frequencies help local communities. It is often the case that poorer communities rent plots of land for agriculture, often miles away. Incorporating a remote soil moisture sensor here allows for the communities only to go to tend to the plots if they need tending. It also provides information as to how the plot of land should be tended to, this provides a far more time and resource-efficient way for the agricultural process which communities have found very helpful. This increased time on the communities' hands has now allowed them to devote more time to education and other activities found beneficial to the community. While this is a small-scale example of the social benefits of implementing such communities, they are instrumental in breaking poverty cycles and increasing the potential for prosperity among communities that most desperately require it.

The above case studies highlight the benefits of IoT and AI. IoT provides modularity from its interconnectivity and therefore provides great flexibility; it is also far cheaper than alternative technologies. The adaptation of IoT has led to many different specialised modules being developed. This means we can take a bottom-up approach in regard to developing larger systems. In other words, we only use the modules necessary for our use case. This is not only cost-efficient but resource efficient, which bodes well for our goals.

AI, on the other hand, provides a lot of leverage in terms of scalability. As Farm Beats is conducting soil nutrient and moisture analysis by AI, there is no longer a need to install more moisture sensors or conduct expensive and time-consuming nutrient analysis per area. Instead, all it takes is more images which are significantly more time effective; the running cost is also much lower even if the startup cost is greater.

Smart agriculture and especially automated monitoring systems for the efficient use of natural resources are not only shown to be viable but as the go-to solution for tackling the issue of depleting resources. It would mitigate the already existing disparity between resource demand and availability and lead to an adaptive strategy should resource disparity reach a point beyond salvaging.

Furthermore, it is not only sustainable in the environmental and social context but economically so too. McKinsey and Co reported in 2020 that increased connectivity in agriculture could unlock $500 billion in GDP by 2030. The article also states, "In North America, where yields are already fairly optimized, monitoring solutions do not have the same potential for value creation as in Asia or Africa, where there is much more room to improve productivity", backing up my claim that the value in these solutions is primarily concentrated in the developing and emerging world. The economic research and successful case studies demonstrate that the solution is not simply an idealist take on solving global issues. Still, it is viable and beneficial for all areas of interest, i.e. social, economic and environmental.

1.3 AI methods for soil pH determination

Due to the fact that accurate electrical pH sensors (which give fast readings) are generally costly, I decided to assess the viability of various machine learning and AI methods for an approximation of pH values. A key inspiration for my approach is the paper "Determine the pH. of Soil by Using Neural Network Based on Soil's Colour" by M Aziz et al., published in the "International Journal of Advanced Research in Computer Science and Software Engineering". [11]

Their proposed methodology involves a three-layered Artificial Neural Network (ANN) Consisting of 3 layers. These layers are: 3 input nodes (for each RGB value respectively), 10 nodes in the hidden layer as well as a single output node, corresponding to the final pH (Appendix B.3). The training of the network involves the standard back-propagation algorithm and using four separate indicators to evaluate performance. The authors have not explicitly specified cost functions, optimisation algorithms, nor if any activation functions were used.

The authors mention that more data should be used to improve the performance of the model, as does a meta-analysis [12] of different AI methods used for a multitude of soil measures. This is understandable as the aforementioned issue with manually assessing soil pH is very time-consuming, and it is known that AI models generally require hundreds or thousands of samples to achieve proper generalisation.

However, I hypothesise that a lack of data may not be the primary issue with any performance setbacks concerning this approach. The data used by the authors is secondary and comes from the paper "Determination of soil pH by using digital image processing technique" [13] provides only an average RGB value from the images taken. This may be problematic as vastly different soils could produce the same average RGB value. Consider the following:

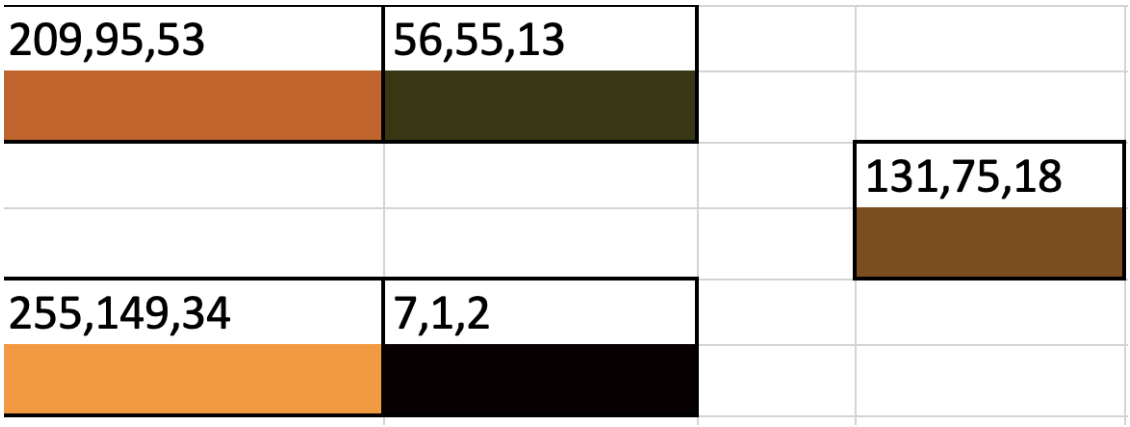


Figure 1.1: Issue with using average RGB

The colours in their respective rows represent two pixels of different colours. Their individual RGB values are noted above. Averaging the RGB values, which are together in a row, gives the RGB value on the right. Both rows average to the same RGB value despite being vastly different colours. The colours on the first row are more likely to come from a coniferous forest, whereas the other one is much more likely from a coastal setting.

To further explore the relationship between RGB values and soil pH as proposed by the authors, I will perform standard data analysis to base my claims further; this is explored in more detail in the methodology chapter.

It is notable that there aren't any papers discussing whether there is an understood relation-

ship between average RGB values and soil pH. However, some research has been done into the HSV colour space and soil pH. A paper utilised the HSV / HSI colour space and performed a logarithmic and quadratic regression to capture the relationship between image saturation and soil pH with a correlation accuracy of up to 84.9% and 86% respectively[14]. I will attempt to replicate this logarithmic relationship between the data set provided by [13] and use the results to inform further actions.

Another reason why I believe that using average RGB values is not sufficient is that it is known that soil pH is a function of: "The rock from which the soil was formed (parent material) and the weathering processes that acted on it—for example climate, vegetation, topography and time" (Queensland Government of Australia). The contributing factors to soil pH will affect not only the colour of the soil but its structure, mineral content, degree of dispersion etc. For this reason, I posit that a model for soil pH should be holistic. It will encompass these factors by considering not only image colour but also intensity, the variation and distribution of pixels in the image and the image texture.

It seems to me that when it comes to developing a complex model for image analysis to predict some feature, a Convolutional Neural Network (CNN) should be the best approach. There seems to be a gap in this respect, as I could not find any CNNs used for the explicit purpose of pH determination. Still, there have been approaches for using a CNN for other soil nutrient analysis [15], which provides precedent to try using a CNN for soil pH analysis.

Since CNNs are capable of using the entire image to create feature maps that may be representative of colour, intensity or even texture, they have the potential to generate a more holistic model of the form $f : image \rightarrow pH$, which would address my key criticisms of the past work in this field. I will therefore endeavour to create such a network and discuss it in further detail in the Methodology section.

Chapter 2

Methodology

2.1 Architecture

I propose splitting up the app into 3 development components which will eventually come together to produce the final front-end user site. These will consist of the Hardware aspect (component 1), the Database and API aspect (component 2), and finally the Machine Learning / AI development section (component 3). Each component should be as lightweight as possible with respect to the final deployed app, such that potential bottlenecks arising from network speeds or hardware specifications are limited. This will be a core principle going forward with the design of the app.

The advantage of this structure is that any component may be changed individually and not break the function of the overall system. For example, if we need to expand the area being monitored but the present microprocessor has all of its pins used up, we can add another separate microprocessor with its own connected sensors and it will integrate seamlessly into the overall system. This architecture also prevents any issues arising from two monitored areas being a far distance away. As each microprocessor module only collects data from a given area and sends it to one main server, it is entirely self-reliant and provides great flexibility for the user's desired set-up.

The same principle applies to the other components, a different AI model will not affect the workings of any other components in the system, same goes for the API and Database.

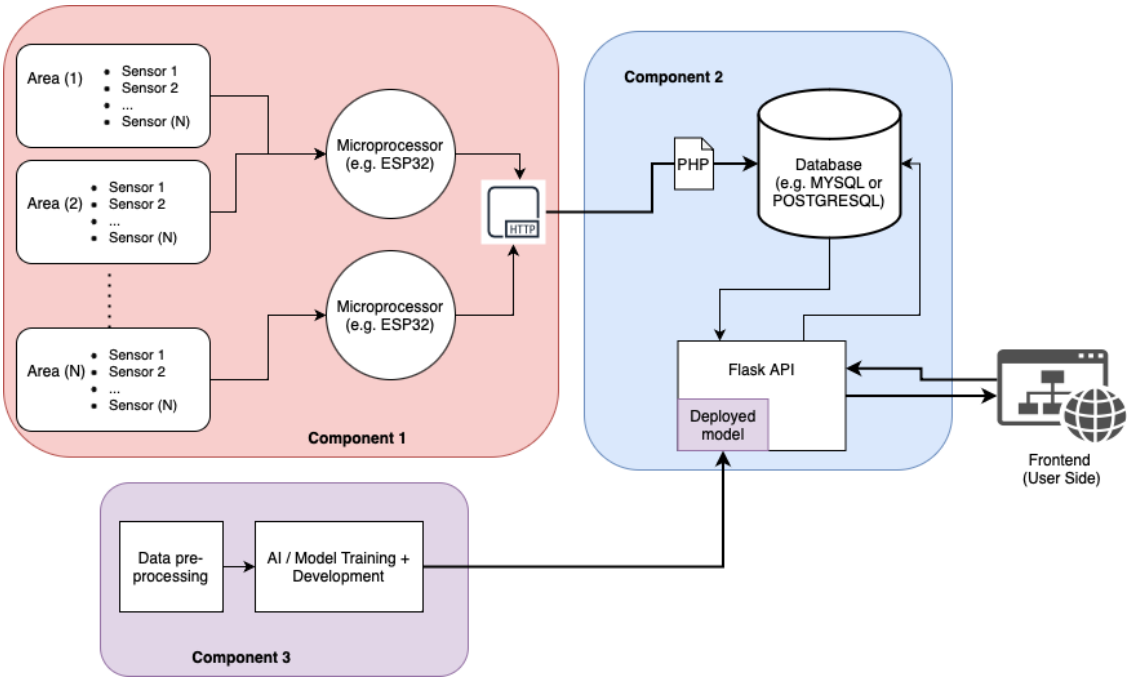


Figure 2.1: High-Level Sequence Diagram

2.1.1 Component 1: Hardware

Each area will have appropriate sensors according to its needs e.g. soil moisture sensors, humidity sensors etc. These sensors will connect directly to a WiFi-enabled Microprocessor, preferably with analogue input signals (as this is generally the conventional output of these sensors). If a particularly large area is being surveyed that requires more sensor data than there are input signals, another microprocessor should be used, as indicated in the Component 1 diagram where some enumerated area (N) is connected to a separate microprocessor. The readings collected by the processors will be sent via an HTTP POST request to the second component of the system.

2.1.2 Component 2: Database and API

Component 2 consists of the Database and Flask-based API. The HTTP Post sub-component should be a simple script to process the post request and insert it into the database (e.g. a simple PHP script), this is to preserve the lightweight-ness of the deployed project as much as possible. This database will store all sensor data in tables separated by location. In addition to sensor data tables grouped by location, there should also be a separate table for soil pH readings which will be found via the AI model hosted on the Flask API. The separation between sensor read tables and pH read tables is necessary as sensor readings will be automatically taken at regular time intervals, while pH readings will be taken at the user's request (shown in the diagram by a line leading from the front-end to the API). Consequently, the respective table schema will be different in areas such as primary keys as well as the automatically updating read times and are therefore not compatible to be in a single table.

I have specifically selected the Flask API as it is Python-based and compatible with industry-standard ML and AI frameworks such as PyTorch and TensorFlow. It is also much more lightweight and explicit compared to other frameworks such as Django and, therefore more in line with the aims of the project.

2.1.3 Component 3: AI / ML

This section is intentionally Isolated from the rest of the components as integrating it would be directly opposed to the aim of keeping the deployed project as lightweight and efficient as possible. Data sets tend to be large, especially ones utilizing images, and the training process for a large model is extremely computationally expensive. Consequently, only the trained and deployed model should be hosted on the deployed app, while all other tasks which go into developing a ready model should be done in isolation. The model hosted will be determined by the results of the AI testing stages discussed further in the paper.

2.2 Hardware Components Used in prototype development

Wireless networks are now widely available even in less economically developed nations. As a result, I have opted to use the ESP32 microcontroller for inputting analogue data. I have opted for the ESP32 for its low cost (approximately $3-4), many input pins (18) allowing for using fewer boards per area, and WiFi capabilities. These attributes make it an appropriate candidate for the board used in the IoT system architecture.

I will use 2 sensor types, namely: a capacitive moisture sensor and a combined temperature-humidity sensor. For the purposes of assessing viability, I will be using 2 of each sensor type such that I may demonstrate the possibility of concurrent data readings.

The data is collected through analogue signals through the pins shown in Figure 2.2, and sent through HTTP Post requests to a remote server. The remote server will process the request via a simple PHP script and insert it into the database (should the data not be corrupted). Figure 2.2 show an example of wiring for the components I listed

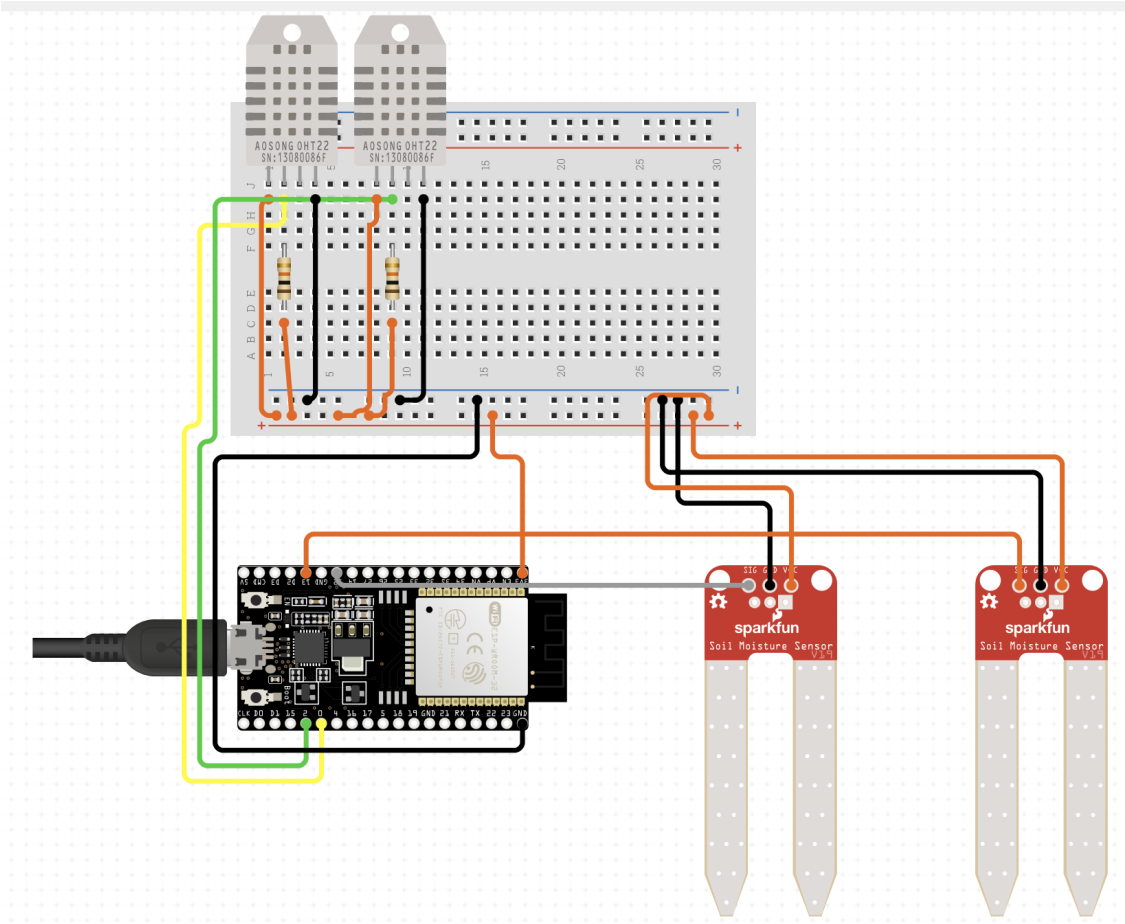


Figure 2.2: Circuit Diagram

(a) Note, the diagram uses different soil moisture sensors to the ones used in the project but provides a better illustration than the alternative. The project used capacitive soil moisture sensors for their superior accuracy, and further details on its specific wiring are found in the appendix

2.3 Artificial Intelligence approach to pH determination

2.3.1 Specific methods

In regard to replicating the neural network in the paper "Determine the Ph. of Soil by Using Neural Network Based on Soil's Colour" by M Aziz et al [11], as the training set is small (30 data points) I will use stochastic gradient descent to prevent over-fitting (due to lack of data as identified in meta-analysis [12]) and make several variations of the network with various nodes in the hidden layer and use cross-validation to determine the best network for the task.

I will also convert the data set to HSV in order to replicate the results in the paper: [16] The paper does not allude to any specific optimizer used, consequently, I conducted some research into optimizers and decided on the Adam optimizer. The Adam optimizer is considered to be very effective for locating global maxima/minima (with respect to the loss function). Research has found that for Multi-layer Neural Networks, "Adam often outperforms other methods" and that "Adam shows better convergence than other methods" [17].

One more common method for increasing model performance is normalising the data to be between smaller scales i.e. $\{0, 255\} \rightarrow \{0, 1\}$ in our case per RGB component. The authors of [11] do not specify if this was done, so I shall try both cases to improve on their achieved performance or get closer to replicating it.

I will also explore whether there is some identifiable relationship between average RGB and soil pH. This will be done through visualisation methods as well as Principle Component Analysis to identify if any combinations of the individual RGB scale components have a direct relationship to pH. To account for the complex non-linear relationship I will repeat the steps with kernalised methods for PCA using all kernels available. The same will be done for the HSV space and the soil index values used by the same paper.

2.3.2 Data Collection

The below lists the sampling locations from which I will take soil samples
(taken from https://www.landis.org.uk/soilscapes/).
I have chosen the location around Chessington for the fact that it is accessible to me and it has the greatest variety of soils of the areas accessible to me.

The specific areas around Chessington were chosen in such a way as to get the greatest variety of soils possible which should maximise the chance of obtaining a good variation in the data set. The legend below the figure provides a description of each soil type and its expected characteristic

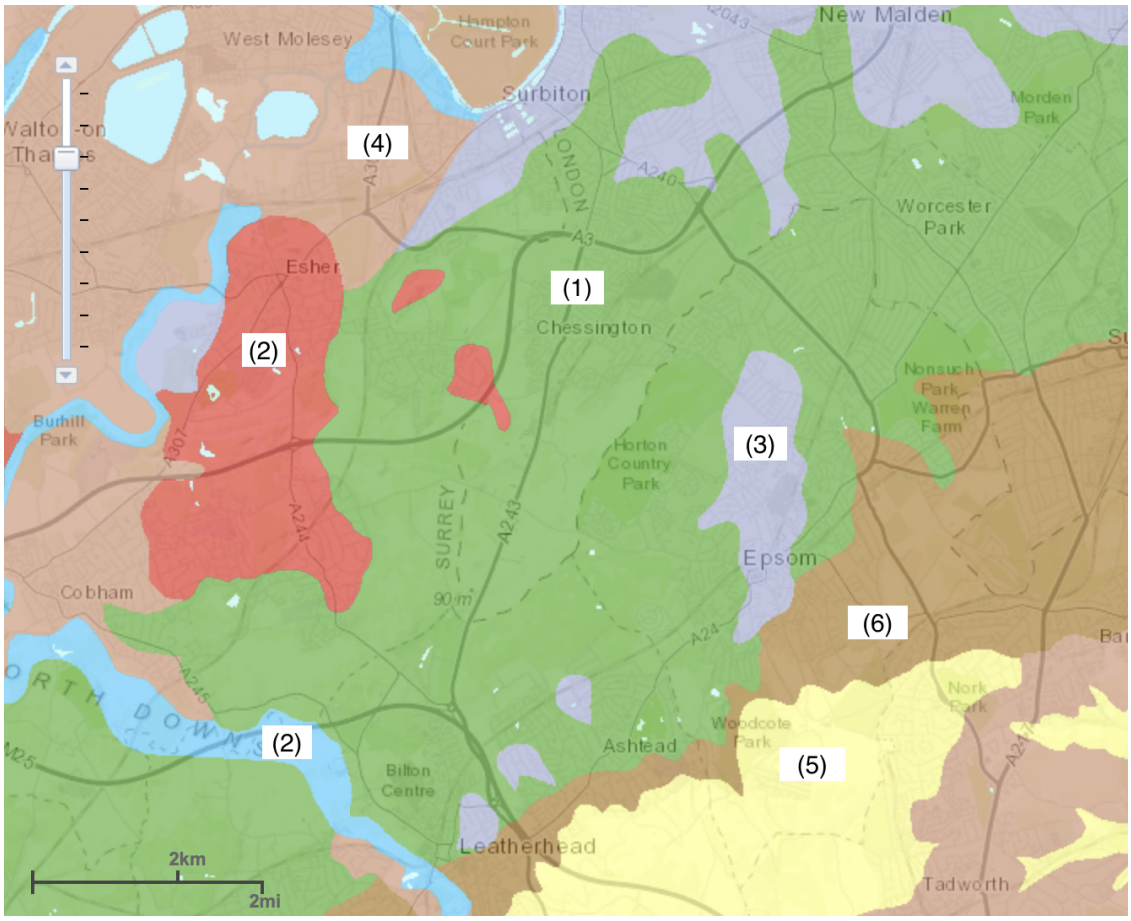


Figure 2.3: Sampling Locations

Colour	Soil Description
	Freely draining slightly acid loamy soils
	Freely draining slightly acid but base-rich soils
	Naturally wet very acid sandy and loamy soils
	Loamy soils with naturally high groundwater
	Slowly permeable seasonally wet slightly acid but base-rich loamy and clayey soils
	Loamy and clayey floodplain soils with naturally high groundwater
	Shallow lime-rich soils over chalk or limestone

(a) Legend

While the number of classes/categories from soil pH will be determined when sampling is concluded. I will aim to split the data into 4 categories to cover the spectrum of moderately acidic to slightly alkaline, which I expect to find in this particular sample area. Re-categorising the data to match these classes should provide more samples per category which is likely to better CNN performance while still providing adequate precision for practical purposes. The categories will

be the following

Categorical Divisions			
acidic	lightly acidic	neutral	alkaline
5.0-5.5	5.5-6.0	6.0-7.5	7.5-8.0

Table 2.1: This division will be referred to by "4-Cat" in AI models

2.4 Data Pre-Processing

I will pre-process the data to improve the performance of the AI models and reduce errors from technological limitations. This will include resizing the images and passing them through a Gaussian filter to remove inevitable noise. After the first two steps of resizing and filtering, I will make multiple variations of the resulting data to test which components of the image are most relevant to model performance. These will include the following image forms and will be done for both the 4-Cat division and the one which results from sampling: Full RGB images (nothing changed from the first 2 steps), HSV images, Edge Detected images (no colour or intensity data) and Grey-scale. For the process of edge detection, I will use Canny Edge Detection and test different thresholds to find the threshold which best captures the texture of the image. This is admittedly a subjective measure, but given the fact that this process has not appeared before in literature, this is what I am limited to.

Following the cleanup of noise and resizing of images, I will normalize their pixel values with respect to the data set in which they are found. This will be done by approximating the mean and standard deviation values for the image data set, and then shifting each pixel for each image in the data set via z-score scaling. The tensor values for the mean and standard deviation can be found in the appendix or appropriate jupyter notebook.

This is a common and powerful technique for increasing the performance of neural networks. The variables used in the pixel shift / z-score normalization will be specific to the variation of the image data set used e.g. HSV, Edge detected etc; consequently the transformations applied to user-uploaded images will depend on the chosen model.

2.5 Convolutional Neural Network

After considerable research into how to design the appropriate neural network for the task. I have opted to use the ResNet-18 model. It is 18-layer deep which should prove adequate for this task due to the relative simplicity of the input data (soil photos). It is a proven CNN architecture which mitigates the vanishing gradient problem and allows networks of many layers to be trained without significantly increased percentage error. This is ideal for this project as I hypothesize that the relationship between images and pH is complex due to many potential components (colour, intensity, texture etc) despite the simplicity of the input data. A complex relationship such as this would require multiple deep layers to capture, which increases the likelihood of the vanishing gradient problem. Using the ResNet model solves this issue.

In testing different models with sample test data, I also found that the time taken to train a ResNet-18 model with the equipment available to me is in fact feasible, this was not the case for some other architectures such as efficient-net or dense-net which proved too computationally expensive for me to train in a reasonable time frame.

Chapter 3

Results and their implications

3.1 Hardware Component

The hardware components cost a total of £9 (not taking into account cables which have a negligible cost per unit) which has a promising start for the affordability goal. This included 2 pairs of capacitive soil moisture sensors and DHT11 humidity and moisture sensors. The ESP32 successfully reads from the sensors' analogue inputs, processes any warnings or issues to do with the readings e.g. critically low moisture, and sends it to a post-data.php file hosted on either a local or hosted server (to be specified by the user for testing) via an HTTP POST request. While the capacitive soil moisture sensors required calibration, the others worked 'out of the box'.

Following an HTTP Post request parameterized by an API token, the ESP32 successfully receives an HTTP response code which in all cases when configured correctly is the HTTP 200 OK.

3.2 Software Component

The PHP file successfully cleans up and processes the request and adds it to a hosted database (which may be changed to a local one by changing url within the file). The Flask app uses a simple MySQL driver to extract the data from the hosted tables and displays them to the user using modern Ajax-based table displays which enable searching and filtering the data as the user prefers. I limited the search function to specific dates and data IDs, as the notification system displays an entry ID if there is an issue with the monitored section, and it is likely a user would like to search specific dates as opposed to specific monitored values.

The notification system also works well both on mobile and desktop, however, it does require the user to give permission, to begin with and, depending on the user's browser and OS, notifications may only appear on the hosted version of the project (hosted currently on "agriproj.me/Flask") as new JS standards specify that notifications may only be sent via a secure context i.e. HTTPS hosts. This can be changed on most browsers' privacy settings.

One not successful aspect of the architecture was hosting the AI model on the hosting service. This is because the hosting service I used to work on the project uses old glib libraries which are not compatible with PyTroch. This sadly means that for this project to fully work on a web-hosted setting would require a migration to a different provider. However, when used locally the AI functions perform exactly as expected. Please follow the instructions in the ReadMe.txt file provided. In the local host of the project, the user is able to select the location for which the soil is being analysed, send an image directly to the site and get a pH range back. The determined pH value is added to the location database. The user may then proceed to analyse another sample or go back to the main dashboard

3.3 Soil pH analysis and ML/AI

3.3.1 Preliminary Analysis of the second-hand soil pH and RGB data

I conducted an analysis to try and find some relationship between average RGB values and soil pH, however, my results suggest that there is no direct relationship at worst and a very complex non-obvious one at best. The covariance and correlation matrices in the notebook 'Analysis_initial.ipynb' indicate no clear relationship between raw RGB values and pH (Appendix B.1). To see if perhaps a combination of the 3 colour channels had a relationship to pH I conducted PCA Analysis. The results of PCA and Kernalised PCA (to account for non-linear relationships) also provided no indicator of an obvious relationship.

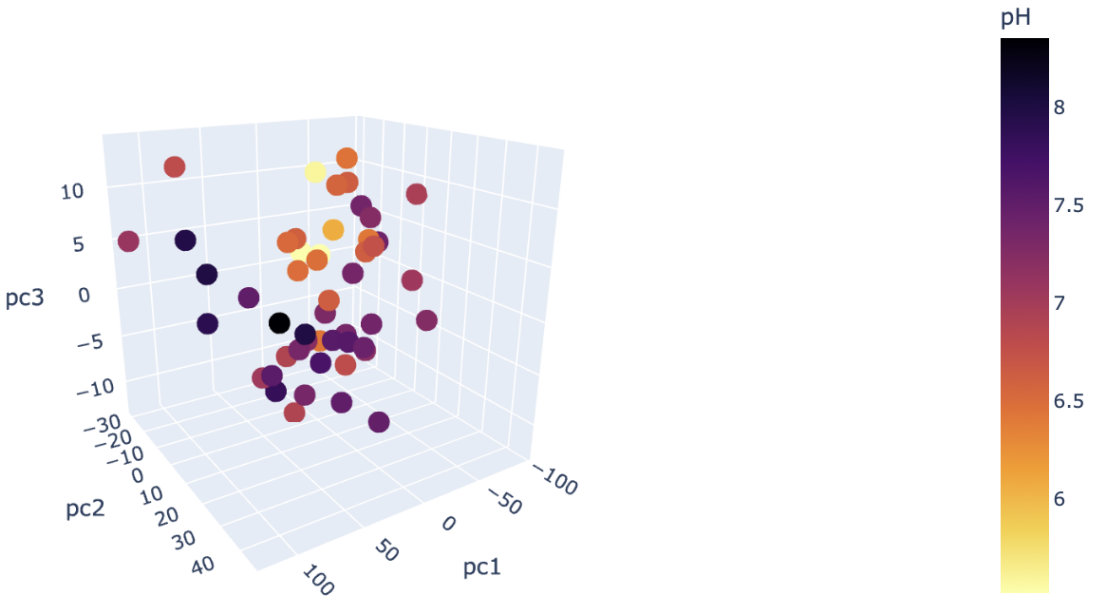


Figure 3.1: PCA components against pH

I converted the data set provided by [13] to HSV / HSI and unfortunately, I was not able to replicate the same relationship using the methods of the authors[14] as well as those used in my previous analysis. This would however make sense, as the locations where the soils were gathered in the two aforementioned papers were from different locations and hence are likely to be characteristically different (data-set used by developers of the HSV/HSI model is not available). Soil index values also didn't show any visual or numerical relationships to pH(Appendix B.2).

Due to the large number of figures which would not fit in this paper, Appendix A contains instructions on finding more figures like 3.1, complete with descriptions and interactive capabilities within Jupyter Notebooks.

3.3.2 Replicating ML/AI models

First I replicated the work of Aziz, et al [11]. As previously mentioned this necessitated some assumptions as the authors did not explicitly state all the details of their neural network (such as any activation functions or training/optimisation algorithms)

In my code titled 'replicated_mlp.ipynb', I created the neural network with the same basic structure as the authors and trained it with MSE Loss and the Adam optimizer. I was able to achieve the same performance as the aforementioned study, however achieving it took an average of 3700

epochs and while the average performance was matched, taking a closer look at the model performance on the test set showed very inconsistent model performance (Appendix Fig B.4). The performance was not any different when using the HSV / HSI version of the data-set.

3.3.3 Collected Data

A total of 80 samples were collected of which 75 were usable. I categorised the images according to their pH level, which yielded 8 pH categories ranging from 5.0-8.0. However, as expected, this categorical division had greatly varying numbers of samples per division. Consequently, I employed the aforementioned divisions (table 2.1). There was still a large disparity between samples in the extremes (very acidic and alkaline) compares to the soils nearer the neutral range (slightly acidic and neutral).

However, in order to get a better idea of whether there is a tangible difference between different pH levels, I also decided to test each model on 8 categories and observe the results. The original class structure will also be used and reported on in case of any interesting results. The full results can be found in the appendix and jupyter notebooks included. The 8-categorical divisions are the following:

Categorical Divisions							
5.0	5.5	5.75	6.0	6.25	6.75	7.5	8.0

Table 3.1: This division will be referred to by "8-Cat" in AI models

One other interesting finding is that the authors of the data set used in the preliminary analysis [13] suggested that more acidic soils are likely to be lighter in colour (more yellowy) whereas more alkaline is likely to have darker characteristics. In my situation, this was not the case. Some more alkaline soils in the 7-8 range likely got their high pH from the chalk and lime sediment underlying the soil and are therefore very bright in colour.

3.3.4 Data Pre-Processing

All images were resized to 256x256 dimensions and had their noise filtered via a Gaussian filter. This filter was appropriate as it removed adequate noise but preserved enough detail in the image such that the Canny Edge detection still performed well in finding edges.

One interesting finding is that when plotting individual RGB components by their quantity for all images, the histograms appeared to be following some pseudo-normal (Appendix B.5) distribution with large outliers towards the extremes. While this was not necessarily relevant for the following discussed results and process, it does become very interesting in later discussion when outlining how I believe the project could be continued and improved on in section 4.6.

All images were normalized after approximating the mean and standard deviation values for the image data set and shifting each pixel via z-score scaling. The tensor values for the mean and standard deviation can be found in the appendix or appropriate jupyter notebook.

3.3.5 Convolutional Neural Networks on variations of data pre-processing methods

Below is a table showing the top 4 Convolutional Neural Network models (based on the ResNet-18 architecture) by prediction accuracy over the different variations of pre-processed data outlined in Section 2.2.3

Top 4 Models			
4-Cat full RGB	4-Cat full HSV	4-Cat Grayscale	8-Cat Texture
100%	93.33%	80%	40%

Table 3.2: Top 4 performing models

Chapter 4

Conclusions and Discussion

4.1 Viability of hardware components

For the purposes of developing a prototype, the hardware components fulfil all needs of the project. They effectively and consistently measure their respective data (soil moisture, humidity and temperature) and send an HTTP Post request over its connected network to the database. Should something go wrong in the Post request or analogue readings from sensors, an appropriate error message is displayed in the output terminal. However, it should be noted that the soil moisture sensor requires normalization at first use. The normalization should take place as follows:

1. submerge a sensor entirely in water and record the output value (max_val)
2. dry the sensor and place it on a dry surface and record the output value(min_val)
3. for every reading (read_val), map the output value between the two previous values. The following equation should be followed

   $(read\_val - min\_val) * \frac{100}{(max\_val)-(min\_val)}$

The other sensors do not require normalization and are ready to go out of the box.

While the individual components work well when it comes to the proposition of expanding the project for industrial use, the cable connections would likely require improvement. The cables used in the project are reusable male-female and male-male cables. These suffice for demonstrating the possibility of such a system, however, for deployment, they should be soldered and should be within a protective plastic casing. This is to reduce the chance of losing a cable connection and improve the system's resistance to water which is necessary for an agricultural project. Luckily such an upgrade would not incur any significant additional costs and therefore the hardware side of the project I consider successful.

4.2 Viability of software component

The software component of the software required many revisions and changes. I decided to preserve the single PHP file to handle the HTTP request from the ESP32 module as it works very successfully provided it is on a hosted (locally or otherwise) server. While initially, I wanted to avoid a heavy API as the functions of the app were computationally simple, hosting the AI component without a Python-based API proved too convoluted.

The Flask API functions well and is expandable for further development. Relative to other popular API frameworks it is lightweight and does not require a lot of storage space or API calls. The readings from hardware are designed to come in every hour making for a maximum of 24

API calls a day. Any subsequent API calls are dependent on the agricultural conditions (as notifications are sent when sensors pick up any problems) and the user themselves when it comes to opening the website and submitting any soil photos for pH analysis.

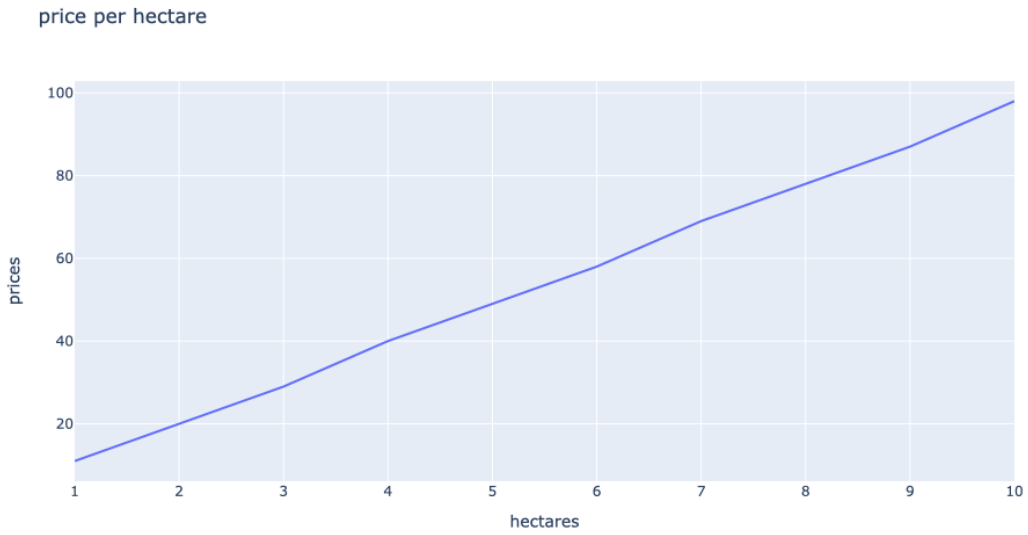
The hosting of the AI model is also successful in principle. The hosting service I used to demonstrate the project had issues with hosting the model due to outdated glib modules, I was only able to find this issue close to the submission date due to the hosting providers longer than average response times, and restrictions to the server command line preventing me from addressing the issue myself out of security concerns on the part of the host. However, on the non-hosted local version of the project, the hosted model works very well. There is a link provided on the website's navbar which allows the user to upload the image, the model normalizes the image according to its previous spec, then returns a pH range to the user and adds it to the appropriate database table. So long as the hosting provider allows for the installation of all necessary models for trained models e.g. in tensor flow or PyTorch this should not be an issue.

4.3 Affordability

The choice of components, and using AI to replace expensive soil pH readers, have made the affordability of this project a great success. The cost will vary according to the size of the area monitored. While many estimates suggest three soil moisture sensors per hectare, the relief of the land could increase this significantly as sudden changes in relief could lead to a non-uniform distribution of soil moisture and would therefore require more than the average 3.

For the purposes of estimating cost I use the values pertaining to this specific project in regard to component costs. I am assuming each hectare requires 2 pairs of the following components (soil moisture, humidity, temperature) as their respective cost and function is representative of real-life agricultural needs. When the size of the farm requires more components than a board has ports then another board is added to the calculation.

Below is a graph displaying the average cost per hectare:



If the average farm is approximately 3 hectares, the cost for the average case is $29. This would be approximately 1.67% of the average resident's income in a low-income country when their GDP per capita is adjusted for purchasing power parity. And when the equivalent calculation is done for lower-middle income countries, it comes out to 0.3% of the average income.

While these figures may not account for someone's debts or unique circumstances and therefore will not be representative for all cases, I believe these figures show that for the average person, this project is certainly affordable and therefore a success in this respect.

4.4 Soil pH analysis and A.I model performance

I believe that the attained results definitively show that a full image performs significantly better than an average RGB value and is more conducive to constructing a model that accurately represents the relationship between image data and soil pH.

The results also imply that the colour data of the image is not the sole indicator of soil pH. Although some relationship is present, other factors must be considered to generate a representative model of the structure $f : image \rightarrow pH$, including but not limited to structure, pixel distribution and intensity.

4.4.1 Preliminary Analysis

While the preliminary analysis showed no distinguishable relationship between average RGB, HSV, and intensity values (even when using advanced kernelized techniques to account for complex relationships), the performance of the convolutional neural networks suggests that colour values have a significant role to play when finding a holistic indicator for soil pH from images. But given the fact that CNNs uses a full image as opposed to different values and reach convergence after a relatively small number of iterations, this is suggestive that the distribution of colours present is more important than the raw concoction of colours.

4.4.2 Convolutional Neural Network Model

The fact the training of the model was computationally feasible suggests that the data preprocessing of resizing was appropriate. Equally Canny edge detection functioned well after Gaussian noise removal and is therefore also likely an appropriate normalisation method. All images were successfully normalized after approximating the mean and standard deviation values for z-score scaling, this technique is also appropriate as it increased model performance in all cases.

The model used cross-entropy loss, and after manual cross-validation, the model containing all RGB data after normalization performs at 100% accuracy. While the dataset is small for a convolutional neural network, lending itself to overfitting, I believe the cross-validation testing and normalization techniques mitigate this sufficiently for the purposes of demonstrating the viability of this model. Images containing the full image data, i.e. RGB, Intensity, Texture, perform the best at 100% accuracy after an average of 101 epochs when the data is divided into 4 categories. This result aligns with my hypothesis that a holistic model accounting for various soil characteristics will likely lead to better model performance due to the fact that soil pH is a complex function of many environmental and chemical factors.

The HSV colour space also performs well, though on average it loses out on some accuracy compared to the aforementioned image space. Due to the fact that the RGB and HSV colour space are interchangeable, I believe this loss in performance is largely due to chance. However, we could reasonably speculate that the HSV-space model could be over-emphasising the texture of the image. Past research has found that the HSV colour space, when edge is detected (as CNN's do implicitly) provides a greater number of edges compared to RGB [18]

It is interesting to note that removing the colour data in place for Gray-scale shows still relatively high performance though not as high as the inclusion of colour data. Implying that perhaps colour itself is a minor indicator of soil pH and instead intensity is far more indicative.

The model which solely takes into account image texture performs rather poorly. It is important to take into account that it uses 8 categories as opposed to the other top-performing models' 4. The 8 categorical texture model however consistently outperformed the 4 categorical equivalent which implies that there is a distinguishable difference in texture between small changes in pH. We must also take into account the fact that though the performance is poor, it is significantly better than the expected performance on chance (12.5%) for 8 categories. The implication of this fact is that the texture of an image plays some small but significant role in indicating its pH, though other data is necessary in tandem to provide a reliable and accurate indicator of pH

4.5 Summary Conclusions

Hardware components used for the project are cheap and reliable. The ESP32 model posts data to the database successfully and does so independently of any other data sources connected to the system i.e. other ESP modules. Given the fact that the hardware platform constitutes the majority of the cost of the project and is still affordable for even those in the most difficult economic situations, I believe I have shown that an IoT/AI solution is viable and effective for combating the problem of lacking resources and increasing the efficiency of agriculture in communities who need it most.

The preliminary analysis showed no distinguishable relationship between average RGB, HSV, and intensity values (even when using advanced kernelized techniques to account for complex relationships). The performance of the convolutional neural networks suggests that the colour and intensity do have a significant role to play when finding a holistic indicator for soil pH from images but only when there are many available per image and not averaged. The CNN, which forms a more holistic model taking into account colour, intensity, texture etc, outperforms all other models and converges significantly faster than attempts to use only average RGB. This is very indicative that average RGB values do not constitute an accurate indicator of soil pH, but my hypothesised model does.

4.6 How the project should be continued and drawbacks

Given more time, I would suggest adding some form of GPS location technology to the ESP32 boards. This could allow for automatically grouping sensors and locations by area rather than having to do this manually. It could unlock the potential to dynamically add tables to the database per location, which would significantly decrease the set-up time and unlock further functionality.

Another issue with the project was the time taken to collect the soil data and, consequently, the lack of it. Each sample takes 30 minutes to process. For 75 samples, this makes for 37.5 hours of sample processing time. Having only 75 images for a CNN in total is very small despite its good performance. To ensure reliability and good generalisation, I would suggest collecting enough data such that each category has at least 25 images. Once this amount of data is collected, I propose creating a generative model to create the rest of the data.

As mentioned in section 3.2.4, following the normalization techniques, the RGB distribution of images seems to follow a normal distribution. This unlocks the possibility of perhaps finding a solution to the difficulty of gathering data for training AI models to determine soil pH. a Generative model such as Variational Auto-Encoder is a neural network that implements a probabilistic bottleneck. What this means is that it creates a latent representation of its training data, as well as a decoder to recover the original image from the latent representation. The advantage here is that the latent representation takes the form of Gaussian distributions (a special case of the normal distribution). Following the training of the model, new data may be generated from the implicitly learned probability distributions for each class in the data set.

Since the distribution of pixel values appears to be Gaussian in nature, this model is certainly appropriate for generating this data, and it is sophisticated enough to capture any relationships

present with respect to the distributions of pixels according to a pH range (Appendix B.6). More advanced models could be used for this purpose such as a Generative Adversarial Model, but I believe in this circumstance a Variational Auto-Encoder is sufficient due to the simplicity of soil images.

Once this generative model is developed, it could be used to generate more data for the ResNet-18 model and consequently improve its performance.

Appendix A

Instructions and further important information

Please find interactive figures, plots, and a detailed showcase of the analysis and AI methods applied within the project in the following Jupyter notebooks in the corresponding file paths: With mkm039/ML as the root path:

- Interactive figures and walk-through of the analysis with runnable code for preliminary analysis of data can be found in 'Analysis_inital.ipynb'
- cost of project per acre approximations may be found in 'quickCalculations.ipynb'
- replication of the Artificial Neural Network may be found in 'replicated_MLP.ipynb'. The saved model state can be loaded by using torch.load_state_dict('replicated_model.pth.tar')

Within the 'mkm039/ML/new_models/' path, You will find notebooks for each different Convolutional Neural Network developed. These involve CNNs without normalization, using HSV, using Canny edge detection, grey-scale etc. The names are descriptive and the user should be able to find them well.

There is also a helper file 'IAAN_Comp.ipynb' which contains code to automatically process files for the neural network should there be new images added:

- recat folder contains re-categorized images into the 4-cat system mentioned in 2.3.2
- ign contains all saved model states tested. The names are also descriptive and may be loaded at the user's convenience
- remaining folders contain different variations of the processed images e.g. 'Normalised' pre-process normalisation images, 'NormalisedGray_C' are grey-scale canny edge detected etc

Appendix B

Figures

	pc1	pc2	pc3	pH
pc1	1.000000e+00	6.252109e-17	-2.051538e-16	0.301876
pc2	6.252109e-17	1.000000e+00	-2.489548e-16	0.197894
pc3	-2.051538e-16	-2.489548e-16	1.000000e+00	-0.407017
pH	3.018760e-01	1.978937e-01	-4.070173e-01	1.000000

Figure B.1: Covariance matrix of RGB PCA

	Soil Index	pH
Soil Index	1.000000	-0.257604
pH	-0.257604	1.000000

Figure B.2: Soil Index Correlation Matrix

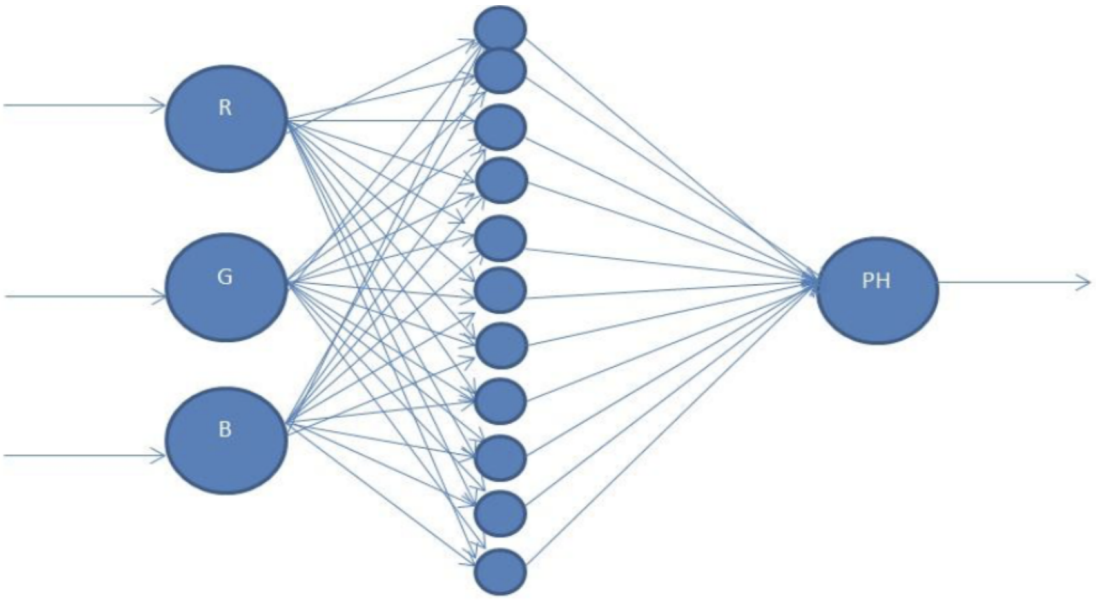


Figure B.3: Aziz et al proposed network [11]

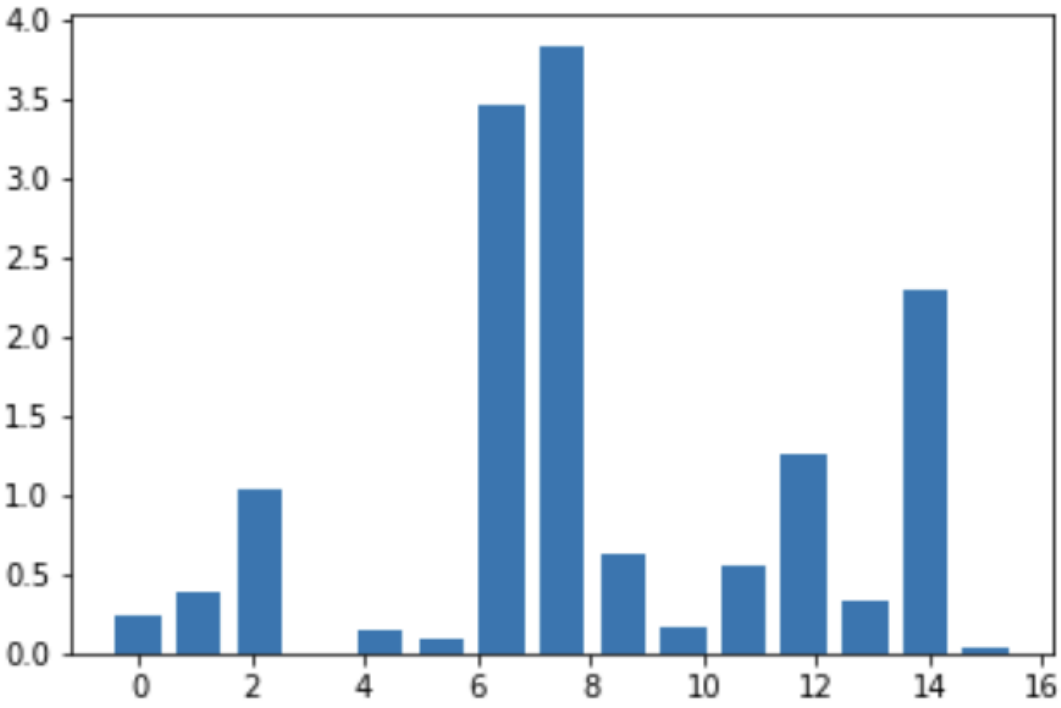


Figure B.4: Replicated Model Performance

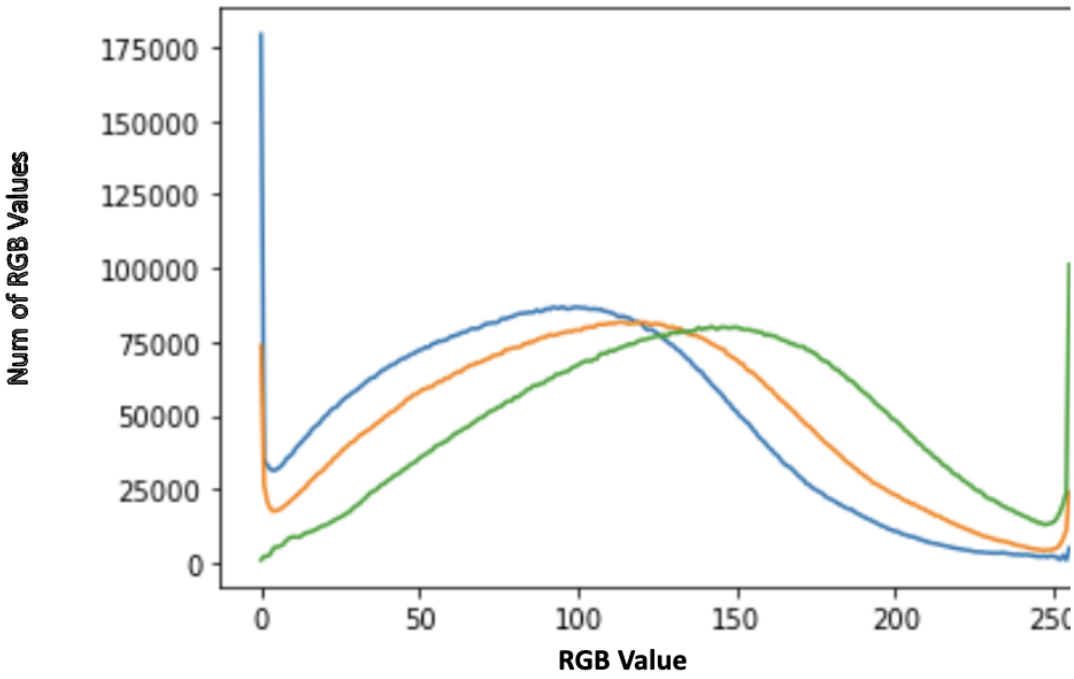


Figure B.5: Histogram of an example soil image

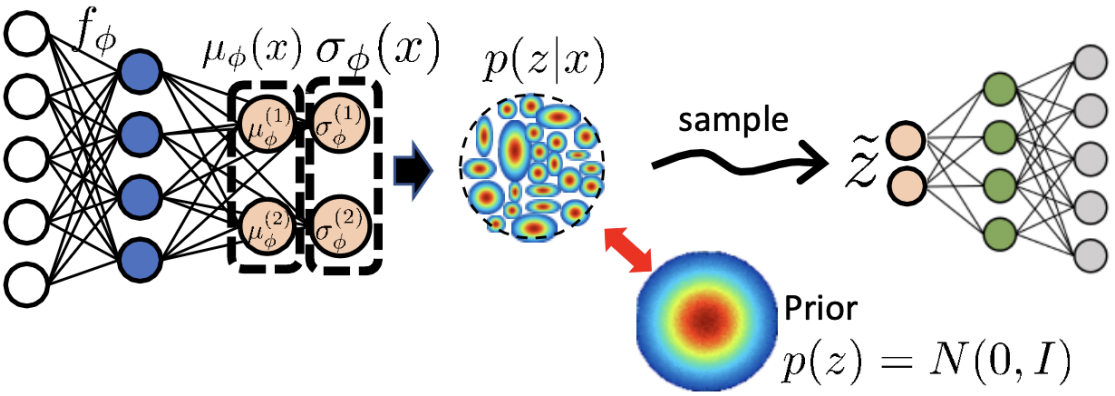


Figure B.6: Proposed Variational Auto Encoder Structure

First layer would consist of soil image inputs, following a bottleneck sequence to give its latent representation of the mean and standard deviation. After the training process, including normalization techniques (D_{KL} divergence) and training the decoder by reconstruction loss. New training samples should be constructed from the $p(z|x)$ distribution

The wiring for the circuits was as follows:

- ESP32 3V to breadboard power terminal
- ESP32 Ground to breadboard ground terminal
- DHT11 sensor 1 to ESP32 GPIO pin 4
- DHT11 sensor 2 to ESP32 GPIO pin 0

- Soil moisture sensor 1 to ESP32 GPIO pin 32
- Soil moisture sensor 2 to ESP32 GPIO pin 33
- Connected all sensors' respective power and ground pins to the appropriate terminals on the breadboard

Bibliography

[1] Daniel Chico, Maite M. Aldaya, and Alberto Garrido. "A water footprint assessment of a pair of jeans: the influence of agricultural policies on the sustainability of consumer products". In: *Journal of Cleaner Production* 57 (2013), pp. 238–248. ISSN: 0959-6526. DOI: https://doi.org/10.1016/j.jclepro.2013.06.001. URL: https://www.sciencedirect.com/science/article/pii/S095965261300382X.

[2] *Population, total*. URL: https://data.worldbank.org/indicator/SP.POP.TOTL.

[3] *population, increase*. URL: https://www.un.org/en/desa/world-population-projected-reach-98-billion-2050-and-112-billion-2100.

[4] Y. Wada et al. "Modeling global water use for the 21st century: the Water Futures and Solutions (WFaS) initiative and its approaches". In: *Geoscientific Model Development* 9.1 (2016), pp. 175–222. DOI: 10.5194/gmd-9-175-2016. URL: https://gmd.copernicus.org/articles/9/175/2016/.

[5] Peter Burek et al. "Water futures and solution-fast track initiative". In: (2016).

[6] Mariana Gonzalez Lago, Brent Jacobs, and Roel Plant. "Panel T04-P02 Session". In: (2019).

[7] Encyclopedia.com. "Saltwater Encroachment, climate change in context". In: (Mar. 2023). URL: https://www.encyclopedia.com/environment/energy-government-and-defense-magazines/saltwater-encroachment.

[8] European Investment Bank. *Wastewater resource recovery can fix water insecurity and cut carbon emissions*. 2023. URL: https://www.eib.org/en/essays/wastewater-resource-recovery.

[9] David Talbot. "Megascale Desalination". In: (Feb. 2015).

[10] *world energy consumption*. URL: https://yearbook.enerdata.net/total-energy/world-consumption-statistics.html.

[11] Makera Aziz, Dena Ahmed, and Banar Fareed. "Determine the pH. of Soil by Using Neural Network Based on Soil's Colour". In: (Nov. 2016).

[12] S. Muthu Saravanan and M. Kamarasan. "A REVIEW ON pH LEVEL DETERMINATION OF SOIL USING IMAGE PROCESSING TECHNIQUES". In: *Journal of emerging technologies and innovative research* (2018).

[13] Binod Vimal, Rakesh Kumar, and Mukesh Kumar. "Determination of soil pH by using digital image processing technique". In: *J. Appl. Nat. Sci.* 6 (June 2014), pp. 14–18. DOI: 10.31018/jans.v6i1.368.

[14] Utpal Barman et al. "Predication of soil pH using HSI colour image processing and regression over Guwahati, Assam, India". In: *Journal of Applied and Natural Science* 10 (May 2018), pp. 805–809. DOI: 10.31018/jans.v10i2.1701.

[15] Muhammad Ammar Jamshed. "Analyze Soil Fertility using Deep Learning Convolutional Neural Networks". In: *Shanlax International Journal of Arts Science and Humanities* 10 (Jan. 2023). DOI: 10.34293/sijash.v10i3.5281.

[16] Utpal Barman et al. "Predication of soil pH using HSI colour image processing and regression over Guwahati, Assam, India". In: *Journal of Applied and Natural Science* 10.2 (May 2018), pp. 805–809. DOI: 10.31018/jans.v10i2.1701. URL: https://journals.ansfoundation.org/index.php/jans/article/view/1701.

[17] Diederik P. Kingma and Jimmy Ba. *Adam: A Method for Stochastic Optimization*. 2017. arXiv: 1412.6980 [cs.LG].

[18] Gwanggil Jeon. "Measuring and Comparison of Edge Detectors in Color Spaces". In: *International Journal of Control and Automation* 6 (2013), pp. 21–30.