Powerline Bushfire Safety Program (PBSP) Vegetation Detection Challenge Background:

In May 2017, the PBSP launched a Vegetation Detection Challenge as there is currently no mechanism that can identify vegetation faults in a timely manner and more specifically what type of species of vegetation is on/touching (causing a fault) a powerline. For example, if a tree or branch touches the line a device could detect the type of tree and send a message back to the distribution business as to whether there is potential for a fire start e.g. urgent action required or that the tree is a peppercorn (for example) and poses no risk, however in time it should be removed from the line.

The Vegetation Detection Challenge aim was for the development of an algorithm that can identify what particular plant species is causing a fault if a tree branch were to fall onto a powerline.

A large amount of Fault Signature detection data collected throughout the PBSP assisted teams with the development of the algorithm.

The Challenge focused on three particular plant species:

- Salix species (Willow) high fire probability
- Franxinus Angustifolia (Desert Ash) medium fire probability
- Schinus Molle (Peppercorn) low fire probability

A consolidation of the fault signature data for the above three species is available on the DataVic website - Vegetation Detection Challenge data. This consolidation removes the need to download the entire fault signature data set.

The full authorised data set including 300GB of photos, videos, test logs and report, are available on the DataVic website.

For a summary of Long Nguyen’s algorithm please see details in the below table:

<table>
<thead>
<tr>
<th>Summary of Approach &amp; Algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td>- This team had a very strong focus on the abilities of machine learning and the Convolutional Neural Network (CNN) algorithm.</td>
</tr>
<tr>
<td>- The team investigated the spectral data (high frequency voltage, high frequency current, low frequency voltage and low frequency current).</td>
</tr>
<tr>
<td>- They determined that the low frequency current data gave the highest level of accuracy of 82% and concluded that the accuracy levels were not sufficient. This was primarily attributed to the lack of sufficient data for training the algorithm.</td>
</tr>
<tr>
<td>- The CNN algorithm was from a standard open source library.</td>
</tr>
</tbody>
</table>
Disclaimer - This material is provided as information only. The Victorian Government and this agency (Department of Environment, Land, Water and Planning):

a) makes no warranty, either express or implied, in respect of the material, including (but not limited to) in relation to the accuracy, reliability, availability, or the currency of the material at any time, or as to its suitability for any purpose; and

b) does not accept any liability to any person for the material, or for the information or advice provided on this website or incorporated into it by reference (or for the use of such material, information or advice). No responsibility is taken for any information or services that may appear on any linked websites.

Please note that Long Nguyen is the owner of intellectual property in the algorithm, which has been licensed to the State in accordance with the terms and conditions of participation in the Vegetation Detection Challenge.

If you wish to reproduce, publish, communicate to the public, adapt, modify or otherwise use the algorithm, you must first contact Long at long.nbl@gmail.com.
hello!

I am Long Nguyen

I am here because I love the innovation.

You can find me at:
https://www.linkedin.com/in/longnb/
1. Artificial Intelligence

Let’s start with the story about a Japanese farmer and skin cancer identification [1][2]
A former embedded systems designer from the Japanese automobile industry named Makoto Koike started helping out at his parents’ cucumber farm.

But he amazed by the amount of work it takes to sort cucumbers by size, shape, colour and other attributes.

Makoto learned that sorting cucumbers is hard and tricky. "Each cucumber has different color, shape, quality and freshness"
Each year in the United States, some 5.4 million new cases of skin cancer are diagnosed.

It’s well established that the earlier the disease is detected, the better the chances of survival.

The five-year survival rate for melanoma detected early on is around 97 percent; but when detected in its later stages, that figure falls to around 14 percent.
What is the common solution for them?
The image recognition
"It’s not hyperbole to say that use cases for machine learning and deep learning are only limited by our imaginations."
The Future Ahead

The artificial intelligence can quickly study a large amount of data and find out the pattern for it to use in a specific purpose.

Artificial intelligence can spot skin cancer as well as a trained doctor

Deep learning algorithm does as well as dermatologists in identifying skin cancer.

In hopes of creating better access to medical care, Stanford researchers have trained an algorithm to diagnose skin cancer.

Google's artificial intelligence can diagnose cancer faster than human doctors

The system is able to scan samples to determine whether or not tissues are cancerous.
Ke Jie is a Chinese professional Go player. As of 24 May 2017, he is currently ranked number one in the world [3].

AlphaGo will now stop playing Go and then throw their energy into the next set of grand challenges, such as finding new cures for diseases, dramatically reducing energy consumption, or inventing revolutionary new materials. [4]

In January 2015, the patient was admitted to a hospital affiliated to the University of Tokyo’s Institute of Medical Science in Japan.

Doctors initially diagnosed her with a type of blood cancer. After treatment, they doubted about their diagnosing.

By cross-referencing the patient’s genetic data with its own database, Watson detected over a thousand genetic mutations in her DNA, in just ten minutes—human scientists would have taken two weeks [5].
Vegetation Detection

It is our turn to try and experiment on the newest technology to solve the old and big problem.
An overview

Black Saturday bushfires [6]

The Director of Energy Safe Victoria told the Commission it was “probably self-evident” that there was an increased chance of fires caused by electrical assets on days of extreme fire danger.

Partial discharge [7]

an electrical discharge that only partially bridges the insulation between conductors and that may or may not occur adjacent to a conductor.
Convolutional network for image classification

Convolution neural network is an artificial network, which has connected neurons like human brain and uses feedforward neural network to move the data in only one direction, forward from input nodes through hidden nodes and to the output nodes.
Japanese cucumber farmer

Makoto spent about three months taking 7,000 pictures of cucumbers sorted by his mother, but it’s probably not enough. Even with this low-res image, the system can only classify a cucumber based on its shape, length and level of distortion. It can't recognize colour, texture, scratches and prickles.

Skin cancer identification

The team ended up with a database of 129,450 images spanning 2,032 different diseases.

The deep neural network then scanned these pixel by pixel, looking for the characteristics common to each diagnosis by showing it thousands of pictures of skin conditions.
Program Algorithm

Vegetation Detection

**Step 1 & 2**

- IND Data
- Analysing

**Step 3**

- MATLAB

**Step 4**

- High Risk Group
- Medium Risk Group
- Low Risk Group

**Step 5**

Vegetation Detection Model
Step by step

Define the data set
Organize and partition the images into training and test subsets

Training the model
Using the deep neural network algorithm to train the model to recognize the different images

Testing and validation
Evaluate the quality of the classifier
System overview
The most important step

The quality of the database will decide the quality of the classifier
The Results

High Frequency Current Database

**Training on single GPU. Initializing image normalization.**

<table>
<thead>
<tr>
<th>Epoch</th>
<th>Iteration</th>
<th>Time Elapsed</th>
<th>Loss</th>
<th>Accuracy</th>
<th>Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>2.95</td>
<td>1.263</td>
<td>35.99%</td>
<td>0.0010</td>
</tr>
<tr>
<td>25</td>
<td>50</td>
<td>137.36</td>
<td>0.0380</td>
<td>90.11%</td>
<td>0.0010</td>
</tr>
<tr>
<td>50</td>
<td>100</td>
<td>374.82</td>
<td>0.0024</td>
<td>98.44%</td>
<td>0.0010</td>
</tr>
<tr>
<td>75</td>
<td>150</td>
<td>511.62</td>
<td>0.0054</td>
<td>100.00%</td>
<td>0.0010</td>
</tr>
<tr>
<td>100</td>
<td>200</td>
<td>549.83</td>
<td>0.0018</td>
<td>100.00%</td>
<td>0.0010</td>
</tr>
<tr>
<td>125</td>
<td>250</td>
<td>686.55</td>
<td>0.0002</td>
<td>100.00%</td>
<td>0.0010</td>
</tr>
<tr>
<td>150</td>
<td>300</td>
<td>821.75</td>
<td>0.0006</td>
<td>100.00%</td>
<td>0.0010</td>
</tr>
<tr>
<td>175</td>
<td>350</td>
<td>954.25</td>
<td>0.0003</td>
<td>100.00%</td>
<td>0.0010</td>
</tr>
<tr>
<td>200</td>
<td>400</td>
<td>1092.46</td>
<td>0.0003</td>
<td>100.00%</td>
<td>0.0010</td>
</tr>
</tbody>
</table>

accuracy = 0.7826
The Results

Low Frequency Current Database

Training on single GPU.
Initializing image normalization.

<table>
<thead>
<tr>
<th>Epoch</th>
<th>Iteration</th>
<th>Time Elapsed</th>
<th>Mini-batch</th>
<th>Mini-batch Loss</th>
<th>Base Learning Loss</th>
<th>Accuracy</th>
<th>Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>2.86 s</td>
<td>1.4300</td>
<td>17.19%</td>
<td>0.0010</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Warning: GPU is low on memory, which can slow performance due to additional data transfers with main memory. Try reducing the 'MiniBatchSize' training option. This warning will not appear again unless you run the command: `warning('on','net_cnn:warning:GPUTooLowOnMemory')`.

| 25   | 50   | 127.28 | 0.0394 | 100.00% | 0.0010 |
| 50   | 100  | 261.07 | 0.0022 | 100.00% | 0.0010 |
| 75   | 150  | 374.49 | 0.0022 | 100.00% | 0.0010 |
| 100  | 200  | 507.48 | 0.0011 | 100.00% | 0.0010 |
| 125  | 250  | 630.48 | 0.0003 | 100.00% | 0.0010 |
| 150  | 300  | 751.81 | 0.0003 | 100.00% | 0.0010 |
| 175  | 350  | 873.50 | 0.0003 | 100.00% | 0.0010 |
| 200  | 400  | 1011.15| 0.0003 | 100.00% | 0.0010 |

accuracy = 0.8561
The Results

High Frequency Voltage Database

Table: Training on single GPU.

<table>
<thead>
<tr>
<th>Epoch</th>
<th>Iteration</th>
<th>Time Elapsed</th>
<th>Mini-batch</th>
<th>Mini-Batch</th>
<th>Base Learning</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>(seconds)</td>
<td>Loss</td>
<td>Accuracy</td>
<td>Rate</td>
</tr>
<tr>
<td>-------</td>
<td>-----------</td>
<td>--------------</td>
<td>------------</td>
<td>------------</td>
<td>---------------</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>3.09</td>
<td>1.2317</td>
<td>35.94%</td>
<td>0.0010</td>
</tr>
<tr>
<td>25</td>
<td>60</td>
<td>142.60</td>
<td>0.0190</td>
<td>100.00%</td>
<td>0.0010</td>
</tr>
<tr>
<td>50</td>
<td>100</td>
<td>274.75</td>
<td>0.0005</td>
<td>100.00%</td>
<td>0.0010</td>
</tr>
<tr>
<td>75</td>
<td>150</td>
<td>405.09</td>
<td>0.0021</td>
<td>100.00%</td>
<td>0.0010</td>
</tr>
<tr>
<td>100</td>
<td>200</td>
<td>536.36</td>
<td>0.0004</td>
<td>100.00%</td>
<td>0.0010</td>
</tr>
<tr>
<td>125</td>
<td>250</td>
<td>671.49</td>
<td>0.0005</td>
<td>100.00%</td>
<td>0.0010</td>
</tr>
<tr>
<td>150</td>
<td>300</td>
<td>812.78</td>
<td>0.0003</td>
<td>100.00%</td>
<td>0.0010</td>
</tr>
<tr>
<td>175</td>
<td>350</td>
<td>942.17</td>
<td>0.0003</td>
<td>100.00%</td>
<td>0.0010</td>
</tr>
<tr>
<td>200</td>
<td>400</td>
<td>1090.53</td>
<td>0.0014</td>
<td>100.00%</td>
<td>0.0010</td>
</tr>
</tbody>
</table>

accuracy = 0.7391
The Results

Low Frequency Voltage Database

Training on single GPU.
Initializing image normalization.

<table>
<thead>
<tr>
<th>Epoch</th>
<th>Iteration</th>
<th>Time Elapsed (seconds)</th>
<th>Mini-batch Loss</th>
<th>Mini-batch Accuracy</th>
<th>Base Learning Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>2.55</td>
<td>1.0920</td>
<td>48.44%</td>
<td>0.0010</td>
</tr>
<tr>
<td>25</td>
<td>50</td>
<td>106.36</td>
<td>0.2188</td>
<td>89.06%</td>
<td>0.0010</td>
</tr>
<tr>
<td>50</td>
<td>100</td>
<td>212.98</td>
<td>0.0149</td>
<td>100.00%</td>
<td>0.0010</td>
</tr>
<tr>
<td>75</td>
<td>150</td>
<td>328.62</td>
<td>0.0071</td>
<td>100.00%</td>
<td>0.0010</td>
</tr>
<tr>
<td>100</td>
<td>200</td>
<td>433.12</td>
<td>0.0010</td>
<td>100.00%</td>
<td>0.0010</td>
</tr>
<tr>
<td>125</td>
<td>250</td>
<td>538.10</td>
<td>0.0010</td>
<td>100.00%</td>
<td>0.0010</td>
</tr>
<tr>
<td>150</td>
<td>300</td>
<td>670.94</td>
<td>0.0002</td>
<td>100.00%</td>
<td>0.0010</td>
</tr>
<tr>
<td>175</td>
<td>350</td>
<td>806.62</td>
<td>0.0004</td>
<td>100.00%</td>
<td>0.0010</td>
</tr>
<tr>
<td>200</td>
<td>400</td>
<td>926.78</td>
<td>0.0004</td>
<td>100.00%</td>
<td>0.0010</td>
</tr>
</tbody>
</table>

accuracy = 0.6738
Low Accuracy Rate of the Deep Learning Model

- The Complexity of Data
- The Number of Samples
- The Difference in Scaling of The Images
The Complexity of Data and Number of Samples

The MNIST dataset

- 70000 BW images in 10 classes
- 60,000 training images
- 10,000 testing images

The CIFAR-10 dataset

- 60000 colour images in 10 classes
- 50,000 training images
- 10,000 testing images
The Complexity of Data and Number of Samples

\[ \text{Desert Ash} \quad \text{Peppercorn} \quad \text{Willow} \]

The High/Low Frequency Voltage/Current dataset:

- 213 colour images in 3 classes
- 169 training images
- 44 testing images
Looking forward

Disadvantage

The disadvantage of the program is the input data and the signal

Future ahead

The algorithm shows a path that we can easily implement in the future when we decide the signal and the input data which we are going to use
Thank you
The road ahead.

A new dawn
From infant to young adult: the birth of artificial intelligence and the current state.
Reference


CREDITS

Special thanks to all the people who made and released these awesome resources for free:

- Presentation template by SlidesCarnival
- Photographs by Unsplash