Thermal Detection for Free Flight

Jake Tallman, David Yu, and Maria Pantoja

Department of Computer Science and Software Engineering, California Polytechnic State University, San Luis Obispo, Ca., USA

*Email: mpanto01@calpoly.edu

Abstract. Thermals are regions of rising hot air formed on the ground through the warming of the surface by the sun. Thermals are commonly used by birds and glider pilots to extend flight duration, increase cross-country distance, or simply to conserve energy. This kind of powerless flight using natural sources of lift is called soaring. Once a thermal is encountered, the pilot flies in circles to keep within the thermal, so gaining altitude before flying off to the next thermal and towards the destination. A single thermal can net a pilot thousands of meters of elevation gain. Estimating thermal locations is not an easy task, pilots look for different indicators like color variation on the ground because the difference in the amount of heat absorbed by the ground varies based on the color/composition, birds circling in an area, and certain types of cloud formations (cumulus clouds). The above methods are not always reliable enough and pilots study the conditions for thermals by estimating solar heating of ground (cloud cover and time year/date) and also the lapse rate and dew point of air. In this paper, we present a Machine Learning based solution to forecast thermals. Since pilots in general record many of their flights locally and sometimes upload them to databases, we use the flight data uploaded to determine where the pilot encountered thermals and together with other information (weather and satellite images corresponding to the location and time of the flight) train an algorithm to automatically predict the location of thermals given as input the current weather conditions and terrain information (obtained from Google Earth Engine). Results show that our model is able to converge on the training and validation set with a loss bellow 1%.

1. Introduction

Glider/free-flight pilots seeking to perform cross country flights rely heavily on thermals to maintain altitude as a paraglider or hang glider does not have any kind of propulsion (thus free-flight). Thermals are regions of rising hot air formed on the ground through the heating of the surface by the sun. To improve flight time for gliders (a fixed-wing aircraft that is supported in flight by the dynamic reaction of the air against its lifting surfaces) and other free flight apparatus, detecting thermals and other forms of lift is fundamental. There are a variety of ways to find potential thermals however they can be unreliable and vary by location and weather. Losing lift on a long flight and being unable to find another source can leave a pilot in a frustrating or dangerous situation needing to quickly find a landing zone. There are only two guaranteed indicators of a thermal; a cumulus cloud and birds or other pilots already soaring and gaining lift [1]. We intend to provide a new way to help pilots predict, with accuracy, thermal locations or lack thereof to catalyse a safe and environmentally friendly flight.

Aside from improving the state of free flight, this research will also help propel deep learning on unique input, since in this research we are experimenting on layering satellite imagery along with visualized weather data as image input to try and predict weather patterns that cannot easily be detected. Up to this point thermal detection has been done with mathematical models. Most real time models rely
on the pilot actively experiencing lift, while prediction models such as those performed by XCSkies, a popular soaring weather prediction website, take in weather readings such as lifted index and thermal index as input to their model [2].

Glider pilots encounter different kinds of lift, however thermals, ridge lift and convergence are the most common. This paper will be focusing mostly on thermals although ridge lift and convergence will come into play as well. Ridge lift occurs when wind strikes a cliff face or ridge line at an orthogonal direction causing an upward force of air. This lift region or lift band can be flown in indefinitely so long as wind conditions maintain. See figure 1 for a visual aid on the lift band.

![Figure 1. a) Ridge Lift b) Thermal Lift.](image)

There is some previous work done on using Deep Learning (DL) in thermal and/or weather changes prediction. In the last decade, image processing has been heavily improved by using Neural Networks, in particular for processing satellite images the network presented in [3] (initially used for cell segmentation) has shown consistently good results. For example, in [4] the authors use a U-Net Neural Network (NN) configuration for precipitation forecasting making highly localized predictions that apply to the immediate future (0-6 hour) with a 1km resolution with a total latency of just 5-10 minutes. Treating weather prediction as an image-to-image translation problem (physics-free). Input to this NN is Doppler Radar and satellite imagery. Another application of U-Net for satellite image segmentation can be found in [5] where a convolutional machine learning model with a modified U-Net structure for creating land cover classification mapping based on satellite imagery is used. Finally, [6] and [7] both utilize a convolutional machine learning model with a modified U-Net structure to create land cover classification mapping based on satellite imagery.

The main goal of this paper is to prove that thermals can be accurately predicted using Machine Learning techniques that incorporate as the input to the neurons: satellite images, terrain information (albedo, surface reflection) and weather information (surface temperature, wind, cloud cover, upward long wave radiation flux) at the time of prediction. To the best of the authors knowledge there is no publication that uses the above data as inputs to NN.

2. Implementation

2.1. Description of Input Data
Thermal detection needs data for terrain, weather conditions, and previous occurrences of thermals on a certain location.

- Flight Data information. To train a neural network (NN), as is the focus of this paper, we need the input data to be labelled for true occurrences of thermals. Naturally there is no database of thermal occurrences for training segmentation models. For this project we needed to create a dataset that used terrain and weather for input with masks of thermal occurrences as labels. To obtain confirmed cases of thermals we use database of previous flights recorded by glider pilots. Pilots record their flights using Track log files in the standard IGC format [8]. IGC is a standardized type of GPS trail record used specifically by free flight pilots. Every second or so (depending on the instrument) the pilot's instrument will log their <time> <lat> <lon> <alt> as a row to the IGC trac file. There are other typically added records to the file such as barometer
reading and task track. Many pilots will then upload their flight files to a database. Leonardo Global Flight Forum is a database that compiles many worldwide competition databases where pilots can publicly upload their flight records for free [9]. The standard for the IGC file is maintained by FAI Gliding Commission [10]. We downloaded all of our flight data for this project from Leonardo Paragliding Forum. As part of this paper we needed to pull features from IGC format. Many files are logged on Leonardo also as KMZ/KML files which is a special from of XML used by google earth. Figure 2 is a visualization of a KML on google earth while Figure 2 (bottom part) shows the data held in an IGC file on Leonardo. The data in an IGC or KML/KMZ file has everything we need to determine when the pilot encountered thermals and where.

Figure 2. IGC track and data displayed on google maps.

- Satellite data. To obtain Satellite Images of the flight region we are using Sentinel-2 MSI: Multispectral Instrument, Level-2A. Sentinel 2 is a satellite mission under to Copernicus Program. Sentinel 2 is comprised of 2 satellites in the same sun synchronous orbit with a revisit time of 5 days with low cloud coverage. The Sentinel-2 data is great for this project because it is easily accessible through earth engine [11], the global data coverage is a shorter time than others such as Landsat, and the data provides important terrain information such as albedo and surface reflectance [12].

- Weather. For weather data as with satellite images we use Earth Engine. The weather-related inputs we decided on are temperature 2m above the ground (gfs), downward shortwave radiation flux (gfs), wind horizontal and vertical components (gfs), upward longwave radiation flux (cfsr), lifted index(cfsr), and high res surface temperature [13]. The datasets we pulled from are GFS, CFSR, and Modis. GFS, Global Forecasting System is a weather prediction model created by the NCEP. GFS provides global coverage of weather forecasts at a resolution of 18 miles. For historical predictions we use predictions of forecast hour 0 meaning they are recorded at the time of the flight. GFS works well because we can get instant weather data at the location of the flight without pulling from nearby weather stations which was our initial approach. We are using GFS for temperature just above the ground, downward shortwave radiation flux and wind. CFSR, Climate Forecast System Reanalysis is another weather prediction system provided by NCEP [13]. "It is a global, high resolution, coupled atmosphere-ocean-land surface-sea ice system designed to provide the best estimate of the state of these coupled domains over this period." CFSR is made of a number of different land and sea models that provide forecasting of various environmental conditions. From this dataset we get Upward long wave radiation flux and lifted index. Lifted index is one of the most important data pieces for relating if there will be thermals at all. There are a few overlapping weather predictions from GFS and CFSR however we prefer GFS because it spans further into the future. Both datasets have the same resolution but only CFSR provides lifted index and upward longwave radiation. MODIS, Modis
is provided by NASA space crafts to give measurements of atmospheric and surface conditions. We are using a Modis data set: Terra Land Surface Temperature and Emissivity Daily Global 1km that is based in part off of the Modis albedo readings [14]. This dataset gives us with very high-resolution surface temperature. We cannot always get this data on the day of the flight as it is a satellite image on an orbit, so we take an average image over one month behind and one forward. This will not give us accurate temperature on the day of, but it should tell us where the changes in the surface temperature take place. This will likely have some redundancy with the albedo readings from the Copernicus satellite.

We finally paint on thermal locations as labels as shown in Figure 3. To train a NN for segmentation we need ground truth images, images with the label of the correct location of the thermal as a mask. This is why we paint on the thermal locations as masks. The way we are preparing data it will be ready to use on any segmentation NN. Most segmentation problems segment out visible regions in an image such as a face or a dog however out network is going to be tasked with finding invisible weather patterns from a layered image. As such it is important that the data, we are feeding in has obvious correlation to thermal locations. The final product are images with 11 layers that we can feed as the input to a NN described next.

**Figure 3.** Thermals Encountered Over Surface Terrain from a) Sentinel-2 Dataset b) MODIS.

### 2.2. Description Network

We decided to use historical data only, not real time for this research. The reason for this is because we can use more reliable data rather than predicted weather conditions that may be subject to higher inaccuracies. We do have significant amount of historic data so for this project we wanted to research the validity of a Deep Learning solution to predict thermals on historic data. The validation of the results on the historic data will help us to later obtain better/more current weather data and transform our current model into one that works with live data and can predict the location of a thermal within a day of the start of the flight. The input will be and image with 11 channels the output will be a probability map therefore the best NN configuration to adapt to our needs is the U-Net. Using a U-Net is a good choice because of the lack of training data, and it also seems to be the choice of most Kagglers that participated in satellite imagery competition. This neural network architecture has revealed to be very good in this situation. U-Nets have an ability to learn in environments of low to medium quantities of training data. U-Nets are also very light weight and train incredibly fast.

Parameters used:

- **Initialization:** He
- **Image size:** U-net was designed for images of size $572 \times 572 \times 3$, our images are $256 \times 246 \times 11$
- **convolution size Filters and padding left same as original**
- **batch size:** 32 To fit on GPU memory.
- **Activation:** Relu
- **Optimizer:** Adam optimizer Over Gradient Descent because it converges faster and decays the learning rate which we found to be useful for training.
- **Loss function:** Soft Dice + Binary Cross Entropy
  - The loss based on the Dice coefficient is commonly used for image segmentation as it allows coping with class imbalance.
- **Augmentation** We use spacial augmentation only. Rotation, translation, flipping, and zoom.
• To mitigate overfitting, we decided to try dropout on the encoding and decoding layers. The results of various dropout techniques are described in the results section.
  o We also use Keras callbacks to implement: Learning rate decay, early stopping if the validation loss does not improve for 10 continues epochs or reaches the desired X
  o dropout - we tried basic dropout, spatial dropout and drop blocks. Drop blocks is a dropout method that drops contiguous chunks from a feature map on a layer instead of random units.
• Upsampling using concatenation
• Downsampling using max pooling

3. Results
The metric used to evaluate the score of our training was the Dice Coefficient, also known as F1 score:

\[ \text{soft Dice Loss} = 1 - \frac{2x - |A \cap B|}{|A + B|} \]

The loss was set to the Binary Cross Entropy Loss (BCE) + the Soft Dice Loss. We did this because with BCE alone the dice loss would stay very high and with soft dice loss alone the model would not converge. A summation of the two losses proved to be the best option. We also experimented with a variety of other losses such as Jacard and weighted dice loss to no greater performance.

\[ \text{loss} = \text{Binary Cross Entropy} + \text{Soft Dice Loss} \]

3.1. Experiments
• Dataset: We are running our network against the custom dataset described in section 2.1. The input images contain 11 layers of satellite and weather images. The truth masks describe the thermals encountered by the pilot flying over the input region on the day of the flight. In order to determine the performance of the model we broke our dataset into training and validation.
• Models: The models we tested against this dataset are the U-net, Unet ++, and a tiramisu CNN. We chose the Unet and Unet++ for their performance in other segmentation problems as well as their popularity in satellite image segmentation.

3.2. Final Results
The idea is to find the best model and hyper-parameters for our custom dataset. Ultimately, we found the U-Net to be the most successful model. We were able to converge very well to the training set and validation set with the U-net while the U-net++ was unable to converge at all. Table 1 shows the training and test dice loss for each model. We include the training error to show that the Unet++ would not converge.

| Base Model       | Training Dice Loss | Validation Dice Loss |
|------------------|--------------------|----------------------|
| UNET Dropblock   | 0.0289%            | 0.0266%              |
| UNET ++          | 68.11%             | 72.11%               |

The U-net converged very well on the training and validation set as can be seen in figure 4. We did not see any signs of overfitting from our validation set using the dropblock dropout technique. Figure 5A) shows the results of prediction tiles from multiple flights. The training shows very accurate thermal prediction except for the last sample which has no thermal, yet the network predicts there to be a thermal here. Figure 5B) shows the prediction on an entire flight randomly picked. This flight is not in the training or validation set it was entirely new. The prediction is colored by certainty, red being very certain and blue/black being no chance. There is some error, however the predictions tend to follow the right locations on the image, we expect these errors to improve when we increase the number of images on the input data.
4. Conclusion and Future Work

In this project, we implemented an end-to-end approach for predicting thermals from a homemade dataset, by combining satellite plus weather data. We implemented a CNN based on the U-net architecture developed by [3], and used Copernicus, NOAA, GFS, and LEONARDO to obtain the input to the NN. We successfully created a custom dataset with thermal truth labels and our proposed approach achieved a loss of ~0.02\% on validation samples.

One major triumph of this research is the dataset that we created. Moving forward we can continue to add to this data and test new methods of prediction on it. Building upon this work we will improve our accuracy on predicting historic thermals then expand our current data set to be able to predict thermal locations in real time. We plan to try to tweak our model to perform segmentation on satellite info while making inferences on weather data. We also plan to try to extend our network with further geometric cues and an uncertainty weighted multitask loss as well as some recently developed techniques [15,16].

References

[1] Perez 2010 Finding Thermals: the five-star system, paragliding spain URL: https://www.paraglidingspain.eu/advanced-paragliding-courses/finding-thermals-the-five-star-system/

[2] Skies X C 2005 XC Skies Soaring Forecast URL: https://www.xcskies.com/

[3] Ronneberger O, Fischer P, and Brox T 2015 U-net: Convolutional networks for biomedical image segmentation CoRR abs/1505.04597

[4] Ulmas P and Liv I 2020 Segmentation of satellite imagery using u-net models for land cover classification arXiv:2003.02899

[5] Agustsson E, Uijlings J R R, and Ferrari V 2018 Interactive full image segmentation CoRR abs/1812.01888 URL: https://arxiv.org/abs/1812.01888

[6] Ivanovsky L, Khryashev V, and Pavlov V 2019 Building detection on aerial images using u-net neural networks, 2019 IEEE 24th Conference of Open Innovations Association (FRUCT) (Moscow, Russia 2019) pp 116-122, doi: 10.23919/FRUCT.2019.8711930.
[7] Gonzales C and Sakla W 2019 Semantic Segmentation of Clouds in Satellite Imagery Using Deep Pre-trained U-Nets," 2019 IEEE Applied Imagery Pattern Recognition Workshop (AIPR), (Washington, DC, USA) pp. 1-7, doi: 10.1109/AIPR47015.2019.9174594.

[8] Forster-Lewis I 2009 Igc File Format Reference and Developers’ Guide 5732009. URL: https://xp-soaring.github.io/igc_file_format/igc_format_2008.html

[9] Paragliding Forum 2005 Leonardo paragliding flight database URL: www.paraglidingforum.com/leonardo/tracks/world/alltimes/.572

[10] Payne T 2008 igc2kmz IGC to Google Earth converter URL: https://github.com/twpayne/igc2kmz

[11] Gorelick, Noel, and Hancher 2017 Google Earth Engine, Planetary-Scale Geospatial Analysis for All URL: https://earthengine.google.com/

[12] E.S. Agency 2000 Copernicus Sentinel-2 URL: https://sentinel.esa.int/web/579sentinel/missions/sentinel-2.580

[13] Saha S, Moorthi S, Pan H, Wu X, Wang J, and Coauthors 2010 The NCEP Climate Forecast System Reanalysis. Bulletin of the American Meteorological Society 91 1015–1057

[14] Wan Z, Hook S, Hulley G 2015 MOD11A2 MODIS/Terra Land Surface Temperature/Emissivity 8-Day L3 Global 1km SIN Grid V006. NASA EOSDIS Land Processes DAAC. https://doi.org/10.5067/MODIS/MOD11A2.006

[15] Agrawal S, Barrington L, Bromberg C, Burge J, Gazen C, and Hickey J 2019 Machine learning for precipitation nowcasting from radar images arXiv:1912.12132

[16] Cao J, Cholakkal H, Anwer R M, Khan F S, Pang Y, Shao L 2020 D2det: Towards high quality object detection and instance segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)