CSCE 5222

Feature Engineering
Project Report

Segmentation of Crops
Group 10

Ikshwak Varma Alluri,
Lokesh Madem,
Vamsi Krishna Reddy
Bogasamudram

1.Introduction

Our goal is to Segment crop fields.

The project tries to segment the crop fields present in the images. Segmentation of cultivated crops is useful not only for food supply chains but also helps farmers understand the growth of their crop. It is also important for various organizations to detect which crops grow in various regions and to forecast the yield of crops to address food supply issues. Segmentation of cultivated crops from non-cultivated fields is our primary goal throughout our project (here, non-cultivated fields include ponds, houses, forests, roads, and barren lands). Segmentation of crops comes with its own problems. As paths present in fields may result in multiple fields within a field, the varying textures of various crop fields due to growth changes in crops must also be considered as a single class of crop field.

Fig 1 : Left one is the given dataset image and right one show our project goal

2.Dataset

There are 18 satellite images (which include 9 given images and 9 validation images given) that show the fields, roads, and woods. The size of each image is 2048 by 2048. All the images are in JPG format.

For ground truth labels, we have used [2] which is an online open-source annotation tool from which we extracted our ground truth images in binary format.

Fig 2: Left one is the given dataset and right image is the annotated image from apeer.com

3.Methodology

Our method is based on a combination of different image processing techniques that involve watershed segmentation, Gabor filters, color-based thresholding, and convolution with our kernel. Our methodology to find the crops from the given images involves the following three processes:

3.1 Pre-processing:

Before applying our algorithm, we have done many pre-processing steps to obtain the best results, but some of them are useful for our use case, which involve grayscale conversion, histogram equalization, histogram stretching, and channel splitting (we used saturation, the value channel from HSV, the green channel from RGB, and a grayscale image for our algorithm) as shown in Fig 3 and Fig 4.

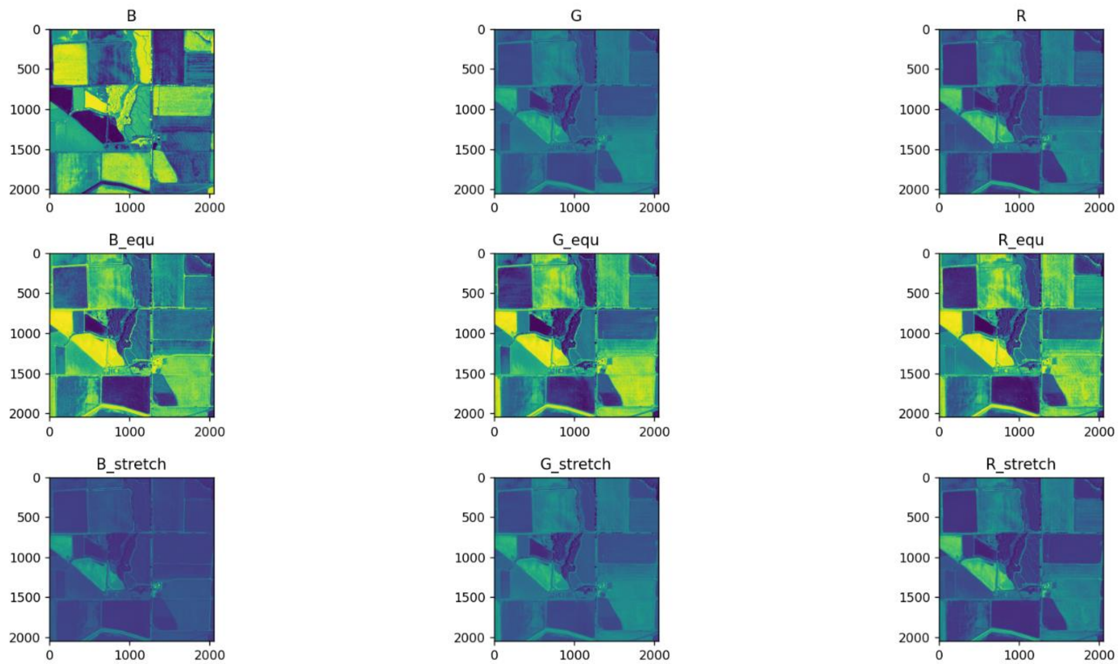

Fig 3: B, G, R channels,
histogram equalization on B,G,R channels,
histogram stretching on B,G,R channels

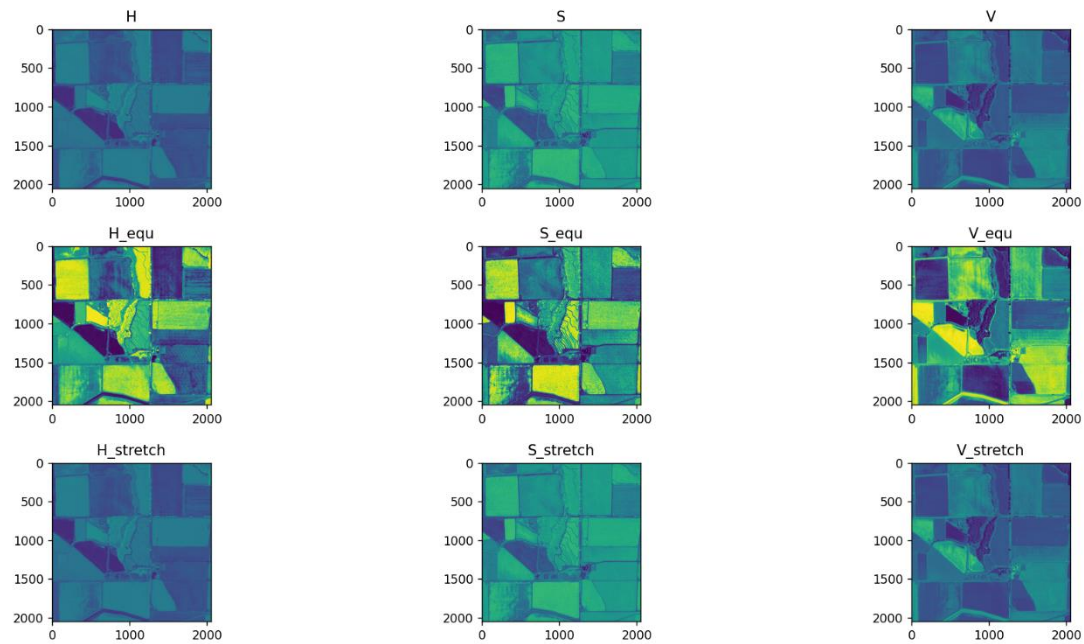

Fig 4: H, S, V channels,
histogram equalization on H, S, V channels,
histogram stretching on H, S, V channels

- We have used the S channel as one of the feature sets for watershed as it includes all the cultivated fields across the entire image dataset, which can be observed in Fig 4.
    - As the forest has a lower intensity, a combination of S and V channels is also used in the watershed, resulting in a more pronounced separation between cultivated and non-cultivated crops, as shown in Fig 4.

3.2 Procedure:

- After preprocessing, we tried the watershed algorithm on the S channel, which recognizes most of the fields. The problem we faced with applying watershed to the S channel is that it is also segmenting forests and barren lands, as shown in Fig 5.

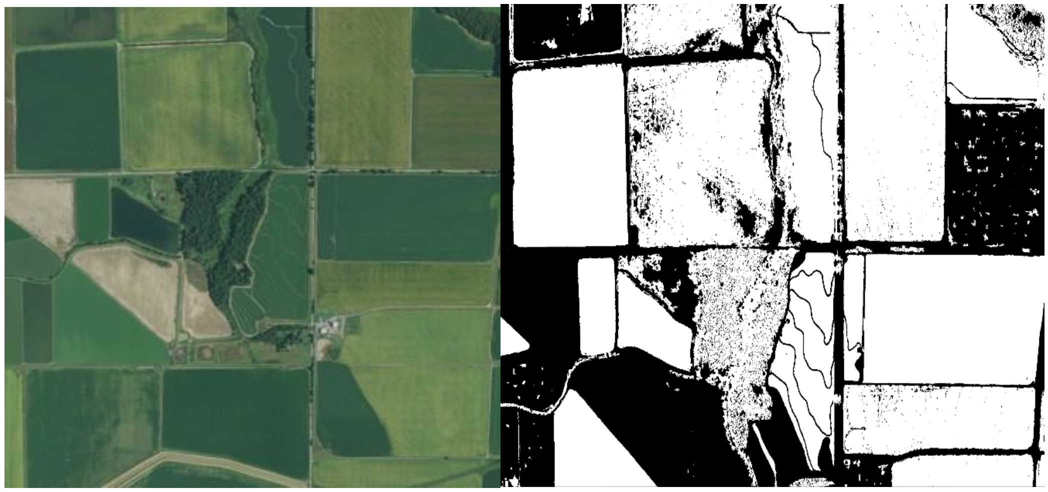

Fig 5: Watershed segmentation on field image with S channel

- So then we took a step back and looked at all the channels and thought that the V channel is separating forests from crops. So we thought that applying the weight sum of S channel and V channel would give the best result, i.e., segmenting crops and removing forests and barren lands. We have used the trial-and-error method for evaluating accuracy. After applying different weighted sums, applying watershed to [(0.67*S) + (0.33*V)] gave the best accuracy. Although the accuracy has greatly improved, this method still recognizes some parts of forests and barren lands, as illustrated in Fig 6.

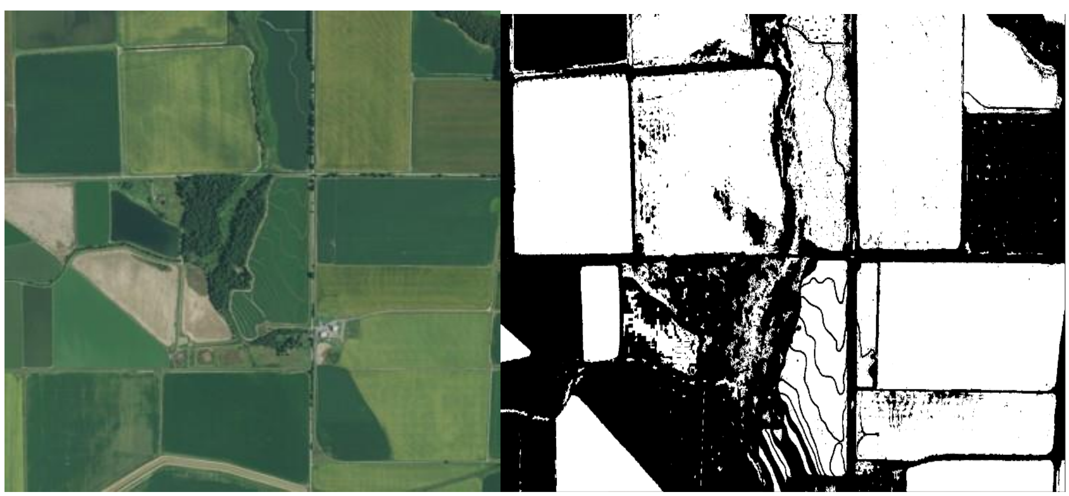

Fig 6: Watershed segmentation on field.jpg image with [(0.67*S) + (0.33 *V)]

- Then we tried K-Means clustering, but it returned lesser accuracy than watershed segmentation. As shown in Fig 7, K-means can segment fields as well as barren lands.

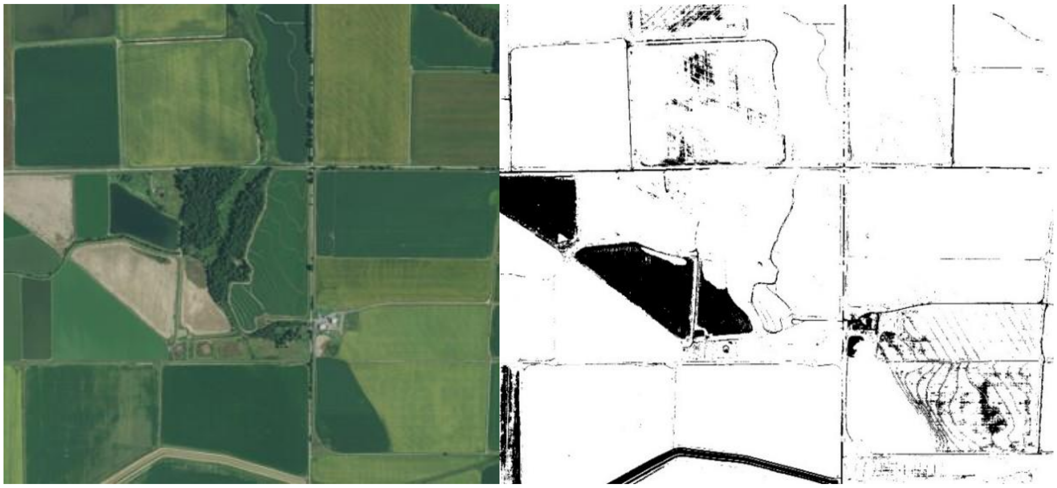

Fig 7: K-Means segmentation on field image

- Hereafter, we thought of applying kernel convolution to the image to recognize forests. We have tried different kernels on the green channel.

Used Kernel = [[1 0 1]][0 0 0] [1 0 1]]

As shown in Fig 8, these kernels were used to try to find the forest in the images, which was obtained through trial and error through observation across all of the images in the dataset.

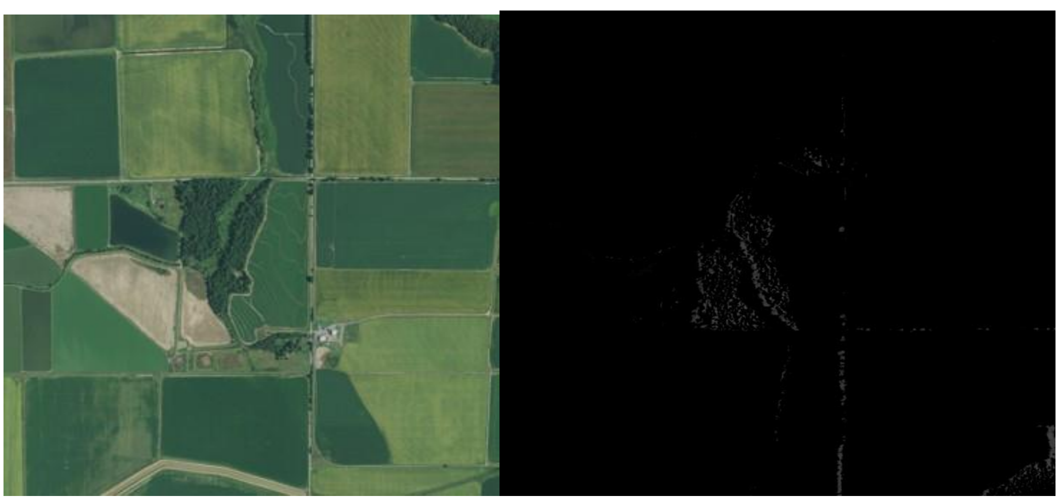

Fig 8: Kernel Convolution on field.jpg image

- At last, we have thought of trying the weighted sum of S and V channels, subtracting Kernel Convolution. After trying different combinations, applying watershed on [(0.67 * S) + (0.33 * V) - (0.4 * conv)] is giving the best results as shown in Fig 9

- For watershed segmentation, we have done thresholding, which involves Otsu-based thresholding. Then we have done morphological operations that involve opening, closing, erosion, and dilation, which result in an image.
- We have performed distance transform on this image, which gives the boundary that is needed for watershed segmentation as a constraint. Initially, for this boundary, we have used Canny-based edge detection as a means of providing boundary, but that resulted in leakage of the watershed, which resulted in poor performance. Then, we have used distance transform in order to get a boundary that did not have any boundary discontinuation, as a result, it gave good performance for segmentation of crops.
- Although we got the fields properly segmented, the forests were also included, which is not what we expected. We have changed different combinations of thresholding and weighted sums, but some of the forests are present in the watershed segmented images, so we try to reduce the forest in the post-processing step through Gabor filters.

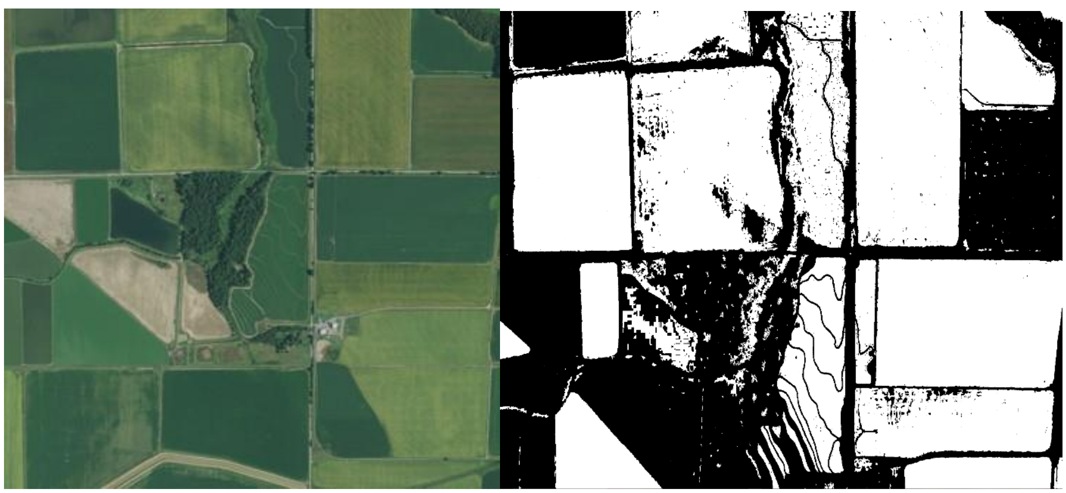

Fig 9: Watershed segmentation on field image with [(0.67*S) + (0.33 *V) - (0.4 * conv)]

3.3 Post-processing:

In order to remove the forests that are present in the water segmented images, we used Gabor filters, which provided us with texture detection of the forests after tuning the parameters such as frequency, standard deviation, and orientation. We have observed that forests are detected across all the images as forests have a unique texture when compared with other forests. We used the built-in Gabor filter in Scikit-Learn.

Gabor parameters used are frequency = 1.6, theta = 45 degrees, and bandwidth = 0.49.

Here, these parameters were obtained through trial and error, where we tried different values and eliminated the ones that were not useful. The above-mentioned parameters are the result that worked better across all the images.

Finally, we subtracted the watershed results from the Gabor-filtered image to obtain our final crop segmentation output, as shown in Fig 9.

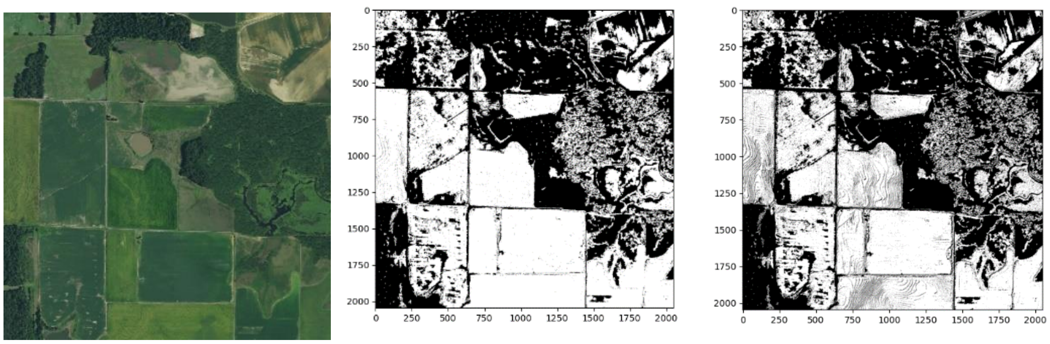

Fig 10: Left one is the original image (L88a.jpg), center image being without using Gabor filter and right image using Gabor filter

- The above results show that forest density is still present but reduced when the Gabor filter is used.
- The right-most image in Fig. 10 also shows that the Gabor filter is used to separate the paths of the field.

Hence, across all the images The Gabor filter in the postprocessing stage improved the overall accuracy and precision, so we included it in our methodology.

4.Experimental Results

For evaluating the performance of our model, we have used a confusion matrix, from which we calculated accuracy and precision through the following formulas:

- Accuracy = "True Positives + True Negatives" or "True Positives + True Negatives + False Positives + False Negatives."
- Precision = (True Positives + False Positives)/(True Positives + False Positives)

The following table shows our results for both the given dataset and validation set.

Table 1 This is an example of a table

Name	Size	Accuracy	Precision
field.jpg	2048*2048	81.8	95.2
L88a.jpg	2048*2048	67.13	24.2
L88b.jpg	2048*2048	79.7	62.7
L96a.jpg	2048*2048	82.9	75.3
L96b.jpg	2048*2048	80.14	87.3
L97a.jpg	2048*2048	87.1	94.1
L97b.jpg	2048*2048	58.09	27.8
W107a.jpg	2048*2048	68.14	87.9
W107b.jpg	2048*2048	71.15	87.4
L88c.jpg	2048*2048	82.02	76.7
L88d.jpg	2048*2048	81.5	72.7
L96c.jpg	2048*2048	77.9	60.4
L96d.jpg	2048*2048	78.3	51.6
L97c.jpg	2048*2048	86.7	90.7
L97d.jpg	2048*2048	83.4	87.7
W107c.jpg	2048*2048	79.2	85.3
W107d.jpg	2048*2048	72.5	97.4

5. Discussion

In the given dataset that we worked with, the combined accuracy is 75.13 and the precision is 71.34, with the maximum accuracy being 87.1 and the precision being 94.1. For the validation dataset given, the average accuracy is 80.2 and precision is 77.84, with the maximum accuracy being 86.7 and precision being 97.4, which shows the performance of our model when segmenting the cultivated crops from the non-cultivated crops.

From above, we can see on average that detection of crops is performed well in both datasets. These findings highlight the significance of our algorithm's performance in detecting cultivated crops.

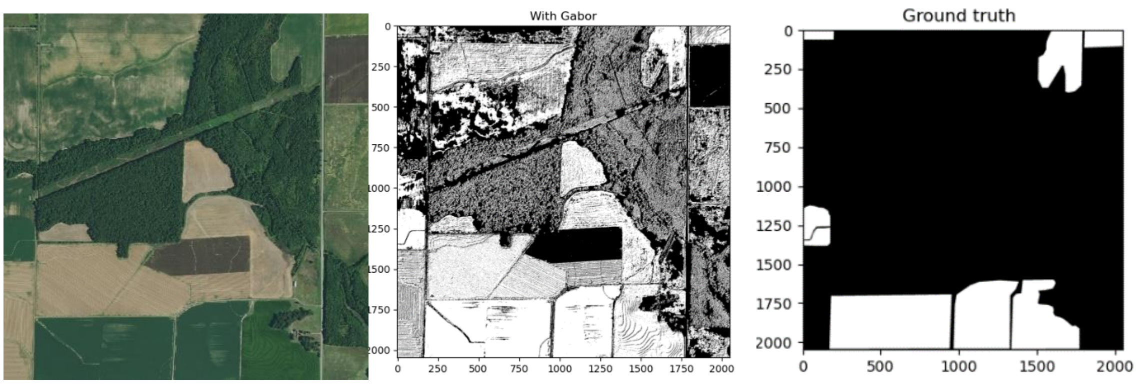

Fig 11: Left most image is L97b.jpg, the middle one being the output of our model and the image in right is ground truth label of the same image

- In such cases, Fig. 11 shows our model not working in cases where it is dominated by non-cultivated lands, as shown in the above image, resulting in the lowest accuracy and precision among the rest of the dataset.
- This leaves us with future work that must be better in such cases where the field images are dominated by non-cultivated lands.

6.Conclusion

Segmentation of crops from satellite images involves many complications when we are trying to separate crops from other objects in the image.

First, the complication comes from the similarity in colors between barren lands and cultivated crops, which makes color thresholding difficult to separate cultivated crops from non-cultivated crops. Also, forests also have similar colors, which clearly shows that color-based thresholding is not enough. Secondly, due to discontinuities at the boundaries, edge-based segmentation algorithms such as Canny, Sobel, and many others did not prove effective in our case. Third, we moved to region-based segmentation, which worked but required many pre-processing steps such as image thresholding, distance transform, and the weighted sum of various channels that proved to segment crops from non-cultivated crops, but it also involves the forests. This process eliminated ponds, houses, and barren lands but could not figure out forests. Finally, we used the Gabor filter as the post-processing step, which for the most part eliminated the forests. Our results show the effectiveness of our model. In conclusion, in this project we tried to segment cultivated crops from non-cultivated crops, for which we presented a novel method.

References:

1. Watkins, B., van Niekerk, A., 2019. A comparison of object-based image analysis approaches for field boundary delineation using multi-temporal Sentinel-2 imagery. Comput. Electron. Agric. 158, 294–302. https://doi.org/10.1016/j. compag.2019.02.009
2. *Automated image analysis*. APEER. (n.d.). Retrieved December 11, 2022, from https://www.apeer.com/home/
3. Zhang et al.,2021, H. Zhang, M. Liu, Y. Wang, J. Shang, X. Liu, B. Li, A. Song, Q. Li
   Automated delineation of agricultural field boundaries from sentinel-2 images using recurrent residual u-net
   Int. J. Appl. Earth Obs. Geoinf., 105 (2021), p. 102557, 10.1016/j.jag.2021.102557