Deep Learning Approaches for Traditional Orchard Detection Using Aerial Imagery

Emilio Ramón Sánchez[a]

[a] *University of Münster, Schlossplatz 2, Münster, 48149, NRW, Germany*

Abstract

Traditional orchards, characterized by diverse fruit trees dispersed across meadows, are vital ecological habitats and cultural heritage in Europe. Despite their ecological significance, reliable data on their distribution is scarce, complicating land cover analyses essential for conservation efforts. This study leverages advancements in deep learning, particularly Convolutional Neural Networks (CNNs), to detect traditional orchards using high-resolution aerial imagery from North Rhine-Westphalia, Germany. By adapting the VGG16 architecture, the research aimed to develop a robust workflow for orchard detection. Although initial results showed potential, challenges such as limited training data and model tuning impacted prediction accuracy. The workflow nonetheless demonstrates promise for broader ecological applications, providing a foundation for future improvements in remote sensing-based habitat mapping.

Keywords: remote sensing, deep learning, land use classification

1. Introduction

Traditional orchards are widespread across many rural regions in Europe, representing both cultural and ecological heritage. They are characterized by a diverse array of fruit trees scattered across meadows, providing microhabitat structures for many species. Their ecological relevance is also reflected in the established German compensatory measurements' system, as their value is consistently rated high (LANUV, 2023). As a result, the creation of new traditional orchard is very popular among construction projects in need to offset their impacts. However, it is quite difficult to obtain reliable information on the number and location of orchards. Some regions have implemented a registry (Äpfel und Konsorten, 2024) but an up-to-date status and completeness are not guaranteed (LANUV, 2024). Reliable information is especially crucial in land cover analysis focusing on ecological value. As ecologist using remote sensing methods like machine learning, analysis is typically based on spectral properties. This is a great technique for most land cover categories and research questions, but it is limited in detecting spatial patterns reliably. Ecosystems like meadow orchards are only detectable through their spatial pattern of trees and grassland. This is also true for many other ecosystems, agricultural contexts or man-made structures. In recent years, advances in deep learning have revolutionized the field of remote sensing and image analysis. Convolutional Neural Networks (CNNs), in particular, have demonstrated remarkable success in various applications, including land cover classification, object detection, and scene segmentation. These models are capable of learning hierarchical features directly from raw image data, making them well-suited for complex pattern recognition tasks. This paper aims to implement a workflow applicable to a variety of research question in the field of ecology.

2. Methods

The deep learning approach for this project was based on the methodology outlined in the tutorial from the OGH Summer School (Knoth, 2020). The usage of freely available data and commonly used programs was a goal of this analysis.

2.1. Data Collection and Preparation

For the detection of traditional orchards, high resolution aerial images created by the federal state of North Rhine-Westphalia were utilized (Geobasis NRW, 2023). These images were obtained for 11 locations to ensure diversity in the training data. Additionally, the official land cover data was used to identify known traditional orchards within the areas of interest (Geobasis NRW, 2024). Binary masks were created using the land cover data and the areas of interest using QGIS (QGIS Development Team, 2023), setting the areas corresponding to traditional orchard to 1.

2.2. Data Preprocessing

The collected images underwent several preprocessing steps to prepare them for training:

- Image subsetting: All images were subsetted into tiles of a uniform dimension of 448x448 pixels to ensure consistency during training.
- Normalization: Pixel values were normalized to a range of 0 to 1 to facilitate faster convergence during training.
- Augmentation: Data augmentation techniques such as rotation, flipping, and cropping were applied to increase the variability of the training dataset and prevent overfitting.

The project was performed using R version 4.4.1 (R Core Team, 2024) and the packages ”keras” (Kalinowski et al., 2024), ”reticulate” (Ushey et al., 2024) and ”tensorflow” (Allaire and Tang, 2024) for deep learning. Additional packages used in the process are listed in the Appendix A.

2.3. Model Architecture

We employed a Convolutional Neural Network (CNN) to detect meadow orchards. The architecture of the model included several convolutional layers followed by pooling layers and fully connected layers. The model's specific layers and configurations were chosen to effectively extract and learn hierarchical features from the input images.

- Convolutional Layers: These layers applied convolutional filters to the input images, learning to recognize various features such as edges, textures, and shapes characteristic of meadow orchards.

- Pooling Layers: Max-pooling layers were used to reduce the spatial dimensions of the feature maps, which helped in retaining the most important features while reducing computational complexity.

- Fully Connected Layers: These layers connected the flattened feature maps from the convolutional and pooling layers to the output layer, performing the final classification task.

In our implementation, we adapted parts of the VGG16 architecture for the specific task. We started with the VGG16 model pre-trained on the ImageNet dataset. This transfer learning approach leverages the knowledge the model has already gained from a large and diverse dataset, allowing it to extract meaningful features from our meadow orchard images.

We utilized the convolutional base of VGG16, which includes the first 15 layers of the model. These layers are responsible for feature extraction. We removed the fully connected layers at the end of the network, as these are specific to the original classification task and do not apply to our task.

After the convolutional base, we added our custom layers to adapt the model to our research question. A Global Average Pooling Layer was added to reduce the spatial dimensions of the feature maps while preserving the important features extracted from the VGG16 layers. It computes the average of each feature map, resulting in a single feature vector for each image. We included fully connected layers to process the feature vector output from the pooling layer. These layers used a ReLU activation function to introduce non-linearity into the model. Finally, we added a sigmoid activation function to the output layer for binary classification (presence or absence of meadow orchards). The output layer produces a single value between 0 and 1, indicating the likelihood of an image containing a meadow orchard.

The model was compiled using binary cross-entropy as a loss function and the Adam optimizer was employed for its efficiency and ability to adaptively change learning rates during training.

The final architecture of the adapted VGG16 model for meadow orchard detection consists of:

- The convolutional base from the VGG16 model (pre-trained).
- A global average pooling layer.
- 14 connected layers.
- A final output layer with a sigmoid activation function for binary classification.

This architecture effectively leverages the powerful feature extraction capabilities of VGG16 while allowing for the customization needed for our detection task.

2.4. Prediction and Validation

The created model was used for prediction on the eleven areas of interest, also partially used for training. A strict train-test-split with a proportion of 80:20 was performed before the creation of the model to ensure good validation. Figure 1 shows the diagnostic plot of the model. Overfitting would be detected here if present.

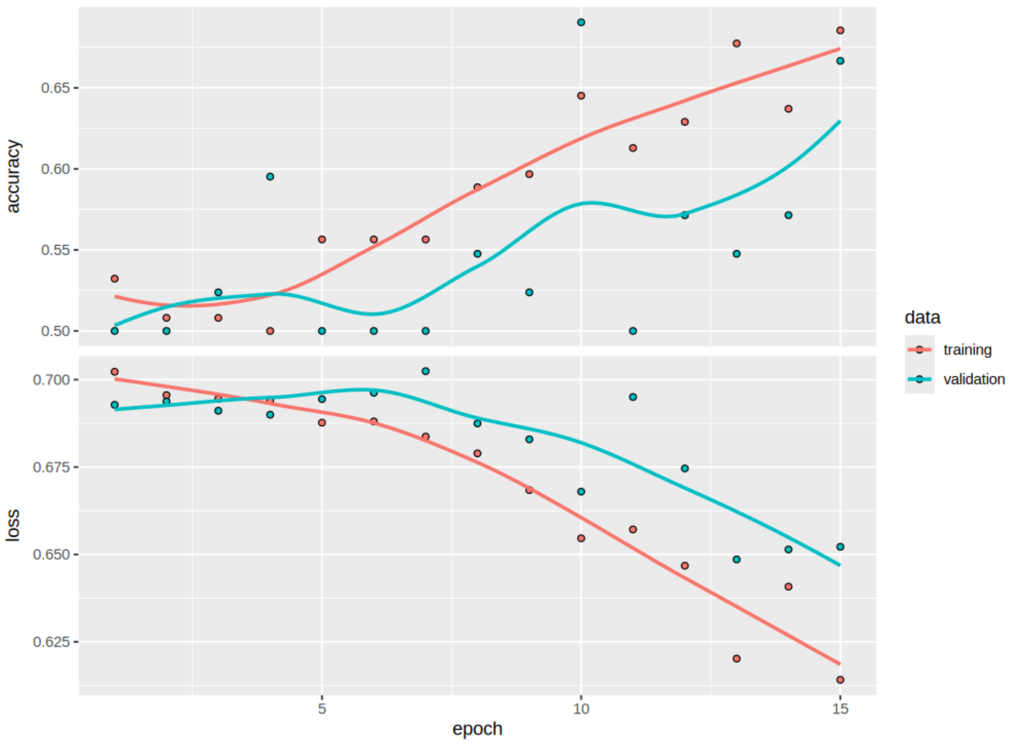

Figure 1: Diagnostic plot of the created model used for the predictions.

3. Results

Presented is the current state of the project with preliminary results, without detailed validation. Figure 1 indicated a fairly good fit of the model and no over-fitting. In total 11 maps were created, showing the predictions in a resolution of 448x448 tiles. In figure 2 the result map for region 3 is shown exemplarily. The red border indicates the location of a meadow orchard. High probabilities are shown in a turquoise colour and low probabilities in a red colour. Most tiles are close to 0.5 indicating a high uncertainty in the prediction. Few tiles have high certainties, however no spatial patterns are visible. Figure 3 shows the first 20 layers of the model applied on image number 802. The edges and structure of the individual trees are clearly visible throughout most layers.

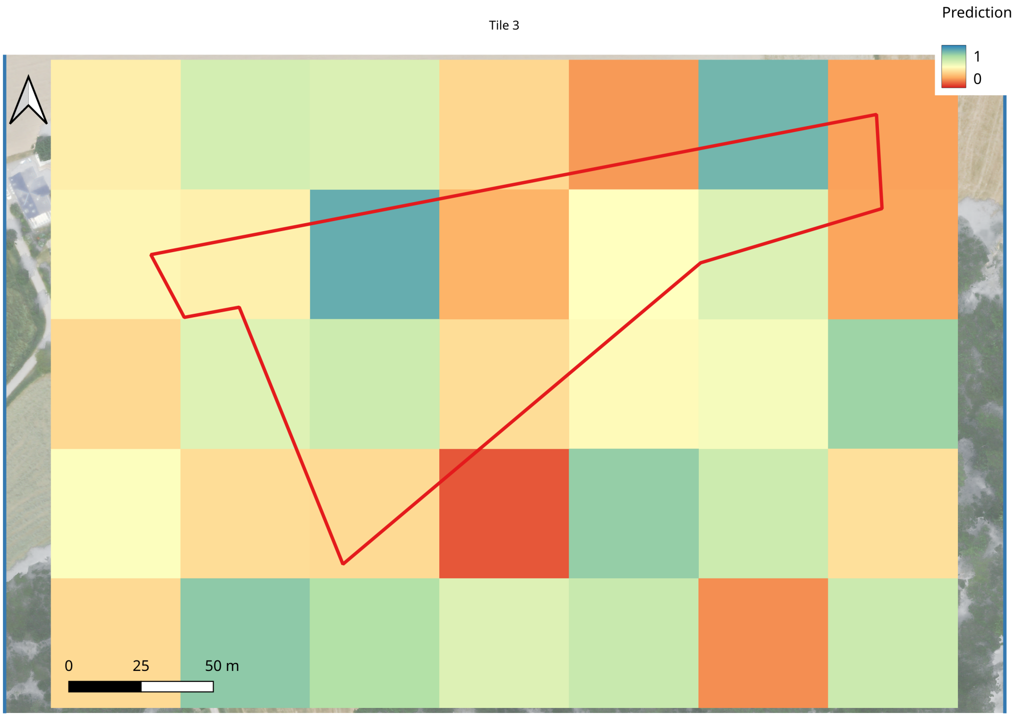

Figure 2: Map of Region 3 with the final predictions for traditional orchards. 1 indicating a high probability and 0 none.

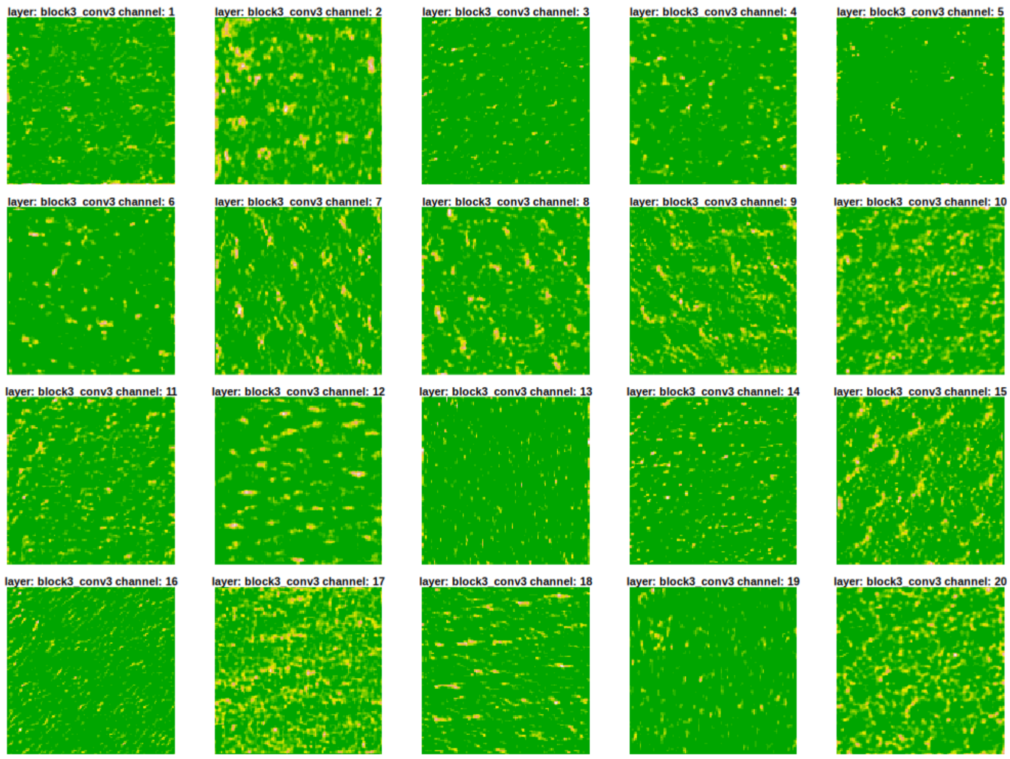

Figure 3: Displayed are the first 20 layers of the neural network, for image number 802.

4. Discussion

The output of the model produces promising results. The prediction maps enable to However, in this application, the results did not yet produce usable predictions. Main reasons for this are the limited amount of time and training data in this project. The main focus was the creation of a workflow, to be used in future projects. This task was successful as we produced prediction maps. Possibly more tuning of the model would result in better predictions. The tile sizes used for the training (here 448x448) and the selection of the input images are prone to subjective selection and the use trial and error procedures. It is likely that the optimal parameters are not yet found, as the detection of meadow orchards is relatively large scale compared to other applications like species detection. Additionally, every model layer and their sequence can be changed or more layers added. Spatial aggregation in a greater scale to reduce the resolution could be helpful. These changes have to be tested to find the perfect fit as the impact can not be foreseen. The use of spatial data also represented a greater challenge than the more typical input of single pictures, as also used in species detection apps like Flora Incognita(Mäder, 2021). Our approach has also impact on broader applications as more accurate and up-to-date habitat maps are possible. This could also influence management decisions or policy making.

In total, the workflow proved to be applicable on typically used and freely available image data. It holds great potential for future application on other research questions and projects. All resources of this project are shared via GitHub to enable reproduction or continuation of the project. Information about the availability can be found in the Appendix.

Appendix A. Full Results

The full code and data is available at: `https://github.com/ESA99/DL_meadow_orchards`. The final prediction maps are also included in the documentation. R packages used in the project are shown in the following table:

Package Name	Version	Citation
knitr	1.47	(Xie, 2024)
gridExtra	2.3	(Auguie, 2017)
ggplotify	0.1.2	(Yu, 2023)
pbapply	1.7.2	(Solymos and Zawadzki, 2023)
neuralnet	1.44.2	(Fritsch et al., 2019)
gtools	3.9.5	(Warnes et al., 2023)
tensorflow	2.16.0	(Allaire and Tang, 2024)
keras3	1.0.0	(Kalinowski et al., 2024)
reticulate	1.38.0	(Ushey et al., 2024)
surveytoolbox	0.1.0.9000	(Chan, 2024)
magick	2.8.3	(Ooms, 2024)
stars	0.6.5	(Pebesma and Bivand, 2023a)
abind	1.4.5	(Plate and Heiberger, 2016)
sf	1.0.16	(Pebesma and Bivand, 2023b)
CAST	1.0.2	(Meyer et al., 2024)
caret	6.0.94	(Kuhn and Max, 2008)
lattice	0.22.6	(Sarkar, 2008)
mapview	2.11.2	(Appelhans et al., 2023)
lubridate	1.9.3	(Grolemund and Wickham, 2011)
forcats	1.0.0	(Wickham, 2023a)
stringr	1.5.1	(Wickham, 2023b)
dplyr	1.1.4	(Wickham et al., 2023)
purrr	1.0.2	(Wickham and Henry, 2023)
readr	2.1.5	(Wickham et al., 2024a)
tidyr	1.3.1	(Wickham et al., 2024b)
tibble	3.2.1	(Müller and Wickham, 2023)
ggplot2	3.5.1	(Wickham, 2016)
tidyverse	2.0.0	(Wickham et al., 2019)
rsample	1.2.1	(Frick et al., 2024)
jpeg	0.1.10	(Urbanek, 2022)
tfdatasets	2.9.0	(Allaire et al., 2022)
terra	1.7.78	(Hijmans, 2024)

References

References

Allaire, J., Tang, Y., 2024. tensorflow: R Interface to 'TensorFlow'. URL: https://CRAN.R-project.org/package=tensorflow. r package version 2.16.0.

Allaire, J., Tang, Y., Ushey, K., 2022. tfdatasets: Interface to 'TensorFlow' Datasets. URL: https://CRAN.R-project.org/package=tfdatasets. r package version 2.9.0.

Appelhans, T., Detsch, F., Reudenbach, C., Woellauer, S., 2023. mapview: Interactive Viewing of Spatial Data in R. URL: https://CRAN.R-project.org/package=mapview. r package version 2.11.2.

Auguie, B., 2017. gridExtra: Miscellaneous Functions for "Grid" Graphics. URL: https://CRAN.R-project.org/package=gridExtra. r package version 2.3.

Chan, M., 2024. surveytoolbox: Useful Support Functions for Survey Analysis. URL: https://github.com/martinctc/surveytoolbox. r package version 0.1.0.9000, commit d1d38701a74323a431eef9527f4074c50b1ce3fc.

Frick, H., Chow, F., Kuhn, M., Mahoney, M., Silge, J., Wickham, H., 2024. rsample: General Resampling Infrastructure. URL: https://CRAN.R-project.org/package=rsample. r package version 1.2.1.

Fritsch, S., Guenther, F., Wright, M.N., 2019. neuralnet: Training of Neural Networks. URL: https://CRAN.R-project.org/package=neuralnet. r package version 1.44.2.

Geobasis NRW, 2023. Digitale orthophotos 2023. Available at: https://www.opengeodata.nrw.de/produkte/geobasis/lusat/akt/dop/.

Geobasis NRW, 2024. Digitales basis-landschaftsmodell. Available at: https://www.opengeodata.nrw.de/produkte/geobasis/lm/akt/basis-dlm/.

Grolemund, G., Wickham, H., 2011. Dates and Times Made Easy with lubridate. URL: https://www.jstatsoft.org/v40/i03/.

Hijmans, R.J., 2024. terra: Spatial Data Analysis. URL: https://CRAN.R-project.org/package=terra. r package version 1.7-78.

Kalinowski, T., Allaire, J., Chollet, F., 2024. keras3: R Interface to 'Keras'. URL: https://CRAN.R-project.org/package=keras3. r package version 1.0.0.

Knoth, C., 2020. Introduction to deep learning in r for the analysis of uav-based remote sensing data. Available at: https://github.com/dachro/ogh_summer_school_2020?tab=readme-ov-file.

Kuhn, Max, 2008. Building Predictive Models in R Using the caret Package. URL: https://www.jstatsoft.org/index.php/jss/article/view/v028i05, doi:10.18637/jss.v028.i05.

LANUV, 2023. Numerische bewertung von biotoptypen für die eingriffsregelung. Available at: https://www.lanuv.nrw.de/natur/eingriffsregelung/numerische-bewertung-von-biotoptypen/.

LANUV, 2024. Streuobstwiesenbestände in Nordrhein-Westfalen -. Available at: https://streuobst.naturschutzinformationen.nrw.de/streuobst/de.

Meyer, H., Milà, C., Ludwig, M., Linnenbrink, J., Schumacher, F., 2024. CAST: 'caret' Applications for Spatial-Temporal Models. URL: https://CRAN.R-project.org/package=CAST. r package version 1.0.2.

Mäder, P., B.D.R.M.S.M.W.H.C.D.A..W.J., 2021. The flora incognita app–interactive plant species identification.

Müller, K., Wickham, H., 2023. tibble: Simple Data Frames. URL: https://CRAN.R-project.org/package=tibble. r package version 3.2.1.

Ooms, J., 2024. magick: Advanced Graphics and Image-Processing in R. URL: https://CRAN.R-project.org/package=magick. r package version 2.8.3.

Pebesma, E., Bivand, R., 2023a. Spatial Data Science: With applications in R. London. URL: https://r-spatial.org/book/, doi:10.1201/9780429459016.

Pebesma, E., Bivand, R., 2023b. Spatial Data Science: With applications in R. URL: https://r-spatial.org/book/, doi:10.1201/9780429459016.

Plate, T., Heiberger, R., 2016. abind: Combine Multidimensional Arrays. URL: https://CRAN.R-project.org/package=abind. r package version 1.4-5.

QGIS Development Team, 2023. Qgis. URL: https://qgis.org.

R Core Team, 2024. R: A language and environment for statistical computing. URL: https://www.R-project.org/.

Sarkar, D., 2008. Lattice: Multivariate Data Visualization with R. New York. URL: http://lmdvr.r-forge.r-project.org.

Solymos, P., Zawadzki, Z., 2023. pbapply: Adding Progress Bar to '*apply' Functions. URL: https://CRAN.R-project.org/package=pbapply. r package version 1.7-2.

Urbanek, S., 2022. jpeg: Read and write JPEG images. URL: https://CRAN.R-project.org/package=jpeg. r package version 0.1-10.

Ushey, K., Allaire, J., Tang, Y., 2024. reticulate: Interface to 'Python'. URL: https://CRAN.R-project.org/package=reticulate. r package version 1.38.0.

Warnes, G.R., Bolker, B., Lumley, T., Magnusson, A., Venables, B., Ryodan, G., Moeller, S., 2023. gtools: Various R Programming Tools. URL: https://CRAN.R-project.org/package=gtools. r package version 3.9.5.

Wickham, H., 2016. ggplot2: Elegant Graphics for Data Analysis. URL: https://ggplot2.tidyverse.org.

Wickham, H., 2023a. forcats: Tools for Working with Categorical Variables (Factors). URL: https://CRAN.R-project.org/package=forcats. r package version 1.0.0.

Wickham, H., 2023b. stringr: Simple, Consistent Wrappers for Common String Operations. URL: https://CRAN.R-project.org/package=stringr. r package version 1.5.1.

Wickham, H., Averick, M., Bryan, J., Chang, W., McGowan, L.D., François, R., Grolemund, G., Hayes, A., Henry, L., Hester, J., Kuhn, M., Pedersen, T.L., Miller, E., Bache, S.M., Müller, K., Ooms, J., Robinson, D., Seidel, D.P., Spinu, V., Takahashi, K., Vaughan, D., Wilke, C., Woo, K., Yutani, H., 2019. Welcome to the tidyverse. doi:10.21105/joss.01686.

Wickham, H., François, R., Henry, L., Müller, K., Vaughan, D., 2023. dplyr: A Grammar of Data Manipulation. URL: https://CRAN.R-project.org/package=dplyr. r package version 1.1.4.

Wickham, H., Henry, L., 2023. purrr: Functional Programming Tools. URL: https://CRAN.R-project.org/package=purrr. r package version 1.0.2.

Wickham, H., Hester, J., Bryan, J., 2024a. readr: Read Rectangular Text Data. URL: https://CRAN.R-project.org/package=readr. r package version 2.1.5.

Wickham, H., Vaughan, D., Girlich, M., 2024b. tidyr: Tidy Messy Data. URL: https://CRAN.R-project.org/package=tidyr. r package version 1.3.1.

Xie, Y., 2024. knitr: A General-Purpose Package for Dynamic Report Generation in R. URL: https://yihui.org/knitr/. r package version 1.47.

Yu, G., 2023. ggplotify: Convert Plot to 'grob' or 'ggplot' Object. URL: https://CRAN.R-project.org/package=ggplotify. r package version 0.1.2.

Äpfel und Konsorten, 2024. Streuobstwiesen-Kataster | Streuobstflächen kartieren in Ihrer Region. Available at: https://www.streuobst-kataster.de/.