methods, data, analyses | 2026, pp. 1-39 DOI: 10.25521/mda.724

# Eye Spy A PSU! Automating Sampling Frame Construction from Aerial Images Using Machine Learning

Adam Eck[1], Trent D. Buskirk[2], Han Shao[1], Peter Stefek[1]

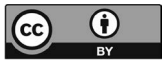
[1] *Oberlin College*

[2] *Old Dominion University*

**Abstract**

The availability of sampling frames is critical for the use of probability-based sampling methods in social science research. Extensive literature addresses how sampling frames can be constructed when the target population consists of people. Less understood is how sampling frames should be constructed when the population being studied consists of places, objects, or locations (POLs). In this paper, we propose an approach that employs machine learning to automate the construction of sampling frames of POLs from aerial (or satellite) images when the POLs of interest have distinctive visual characteristics (e.g., windmills, playgrounds, religious centers). Automating this process with machine learning alleviates the time and monetary costs of researchers manually reviewing potentially several thousands of aerial images to identify sampling units. We evaluate our approach using a case study constructing sampling frames of windmills as POLs within the state of Iowa. We train convolutional neural networks to identify windmills within aerial images from the U.S. Department of Agriculture's National Agriculture Imagery Program and find that our approach successfully predicted 80% of the windmills in the area of interest (1,521 out of 1,913 windmills across ten counties in Iowa) and ruled out 99% of locations lacking the POL of interest (out of over 300,000). Thus, we achieved good coverage in the resulting sampling frames and suggest that any over-coverage could be removed with manual review of only a small number of images — rather than all of them — representing an approximate 98% reduction in the manual effort required without machine learning.

*Keywords*: sample frame construction; places, objects, or locations (POLs); convolutional neural networks; machine learning; survey informatics



Social scientists across many disciplines commonly rely on data collected from samples of a population to make inferences about a total population of interest. Interacting with smaller samples enables more efficient data collection with far fewer monetary and time costs, while the errors introduced by sampling can be minimized with the use of a carefully curated sampling frame and probability-based sampling methods. Familiar examples include election polling, official statistics such as economic indicators, and measures of public opinion. Many sources exist, from governmental records to vendor companies, that offer high-quality sampling frames providing appropriate coverage when the target population consists of various groups of people.

At the same time, an increasing number of research projects are emerging that focus on other types of populations, including places, objects, and locations (henceforth referred to as POLs). For example, economists and environmental scientists might want to study the location of windmills within a state, coupled with the energy production of those windmills, in order to better understand the availability of clean energy sources and their impact on the environment. Similarly, public health officials and psychologists studying education and childhood development might want to understand the availability of playgrounds within census tracts or other geographic areas, for which existing frames rarely exist.

The POLs themselves, or their locations, could also serve as primary sampling units (PSUs) within the first stage of a multi-stage cluster sampling design. For example, windmill locations could first be sampled to represent communities of residents who live near clean energy infrastructure, then residents within those communities could be sampled (from an address-based sampling frame) to better understand the impact of clean energy infrastructure on local economies, lifestyles, and environmental effects. Or pickleball courts could be sampled, and houses within a 1,000-foot radius of these courts could be sampled for a study interested in the effects of possible sound pollution emitted from the courts on households close to these courts. Similarly, public health officials and psychologists might sample the location of playgrounds within municipalities, then interview families utilizing these playgrounds in order to better understand community risk factors for childhood disease and behavioral issues.

Indeed, prior work has demonstrated that sampling frames for specific types of locations (e.g., schools, houses, or buildings) can sometimes be created from vendor or administrative data sources, prior censuses, or registers. These locations may be of primary interest themselves or may serve as a higher-level sampling unit within the context of a multi-stage sampling design. For example, Adamu (2025) created a sampling frame of schools from administrative records

*Direct correspondence to*
Adam Eck, Oberlin College, Oberlin, OH, USA
E-mail: aeck@oberlin.edu

from a state ministry of education in Nigeria. Brummet et al. (2014) used commercial vendor information about schools and teachers to enhance school-based sampling frames for education studies within the U.S. The Health Information National Trends Survey (HINTS) selects households as part of a two-stage design using a sampling frame of addresses obtained from a commercial vendor (Nelson et al., 2004). The Commercial Buildings Energy Consumption Survey (CBECS) creates a sampling frame of buildings for its first stage of selection using multiple sources, including commercial building databases and administrative records, among others (EIA, 2022). In this work we seek to develop a methodology for developing sampling frames for other locations, places, or objects (POLs) that are not commonly included in vendor data or other similar sources.

Frequently, POLs of the types identified above have visual characteristics that make them easily distinguishable (e.g., three long white blades on a windmill) by people. Unfortunately, how to best construct sampling frames for POLs is less understood and often relies on expensive manual processes (e.g., people looking through many images that might contain POLs) that increase the monetary and time costs of the project, making such studies difficult to conduct. For example, McMahon et al. (2014) used high-resolution satellite imagery to identify and manually count individual elephant seals on Macquarie Island, a remote and difficult-to-access location. Their counts were highly consistent with prior, on-the-ground estimates but required human coders to accomplish the enumeration. Chabot et al. (2018) developed a repeatable process that incorporates object-based image analysis using third-party software to detect and count lesser snow geese in large numbers of images of breeding colonies across the Canadian Arctic. Their process produced population counts that were very consistent with manual counts but reduced human effort by an estimated 90%. While such commercially available software might sometimes be used by survey researchers to detect POLs, the off-the-shelf software may not offer flexibility to the user regarding the types of data it can process or the types of places, objects, or locations it has already learned versus what is needed for a given study.

In this paper, we report the results of a research study that explores the use of machine learning methods to aid in the construction of sampling frames of POLs. Specifically, we train deep learning models to predict whether a given aerial image (similar to satellite images viewed in popular web resources such as Google Maps) contains a particular POL of interest. These models learn from examples of images that do and do not contain the type of POLs of interest provided by the social science researcher. Such models can be employed to search through a series of aerial images that cover a wide geographic area of interest to predict all instances of that type of POL, which then serve as the sampling frame from which researchers can draw random samples to study a larger population. Our automated process also aims to balance cost and coverage of sam-

pling frames, compared to its manual counterpart. The two research questions addressed are:

- Research Question 1: Can machine learning models be trained to successfully identify the POLs of interest within aerial images to automatically construct sampling frames for social science researchers?
- Research Question 2: Can we improve the coverage of sampling frames automatically constructed by our process by providing additional contextual information in the models?

Although we primarily focus on sample frame construction in this paper, a key byproduct of achieving perfect coverage in our sampling frame (positively answering Research Questions 1 and 2) is a complete *census* of all POLs within the geographic region of interest. From this census, population count estimates could be derived with reduced monetary and time costs compared to current methods. This complete enumeration could also enable further measurement of properties of every POL besides their location and enable the production of official statistics related to the POLs of interest. For example, Parhar et al. (2022) used just under 2,000 satellite images and two machine learning models to identify the location of solar panels within a smaller geographical region. They also applied an additional model that utilized output from the machine learning models to estimate the surface area of each of the identified solar panels, which could in turn be used to generate official statistics around the density of alternative energy sources within an area or to estimate the solar-voltaic energy production capacity of a geographic region.

**Case study:** As a case study to answer these research questions, we evaluate the quality of sampling frames of windmills in the state of Iowa constructed by our approach. The state of Iowa has the largest per capita collection of windmills in the United States (American Wind Energy Association, 2017). We use the ten counties with the largest number of windmills (comprising 50% of the total windmills in Iowa) as a target application area, for which we construct a sampling frame of windmill POLs and evaluate its coverage. The remaining counties (and their 50% of total windmills in Iowa) are used as training data to construct our machine learning models. The aerial images used in our study for all counties come from the U.S. Department of Agriculture's freely available National Agriculture Imagery Program (NAIP)[1], and we used the U.S. Wind Turbine Database (USWTDB)[2] as the gold standard of the locations of windmills for our total population. We employ convolutional neural networks (CNNs), the state-

[1] General information about the NAIP can be found at https://naip-usdaonline.hub.arcgis.com. The images can be downloaded from: https://datagateway.nrcs.usda.gov/GDGHome_DirectDownLoad.aspx

[2] Publicly available online at https://energy.usgs.gov/uswtdb/

of-the-art deep learning approach for learning from image data (Goodfellow et al., 2016), as our machine learning models. We compare two different types of CNNs that rely on different amounts of contextual information: (a) image classification models, such as ZFNet (Zeiler & Fergus, 2014) and VGG-16 (Simonyan & Zisserman, 2015), learn to make binary yes/no predictions about whether an image contains a POL of interest, whereas (b) image segmentation models, such as U-Net (Ronneberger et al., 2015), additionally learn to predict where a POL exists within an image (if it is present). We hypothesize that:

- Hypothesis 1: Both types of CNNs will be able to accurately discover the locations of windmills in order to automatically construct sampling frames with good coverage.
- Hypothesis 2: Image segmentation models that exploit additional context will result in higher quality sampling frames, with the tradeoff that they will require more elaborate training examples provided by human researchers.

We evaluate the quality of the sampling frames constructed by our machine learning models in terms of their coverage: (a) whether POLs of interest are properly identified; (b) whether the sampling frames suffer from *over-coverage*, when other types of POLs are included in the sampling frame even though they do not belong; and (c) whether the sampling frames suffer from *under-coverage*, when POLs of interest are missed by the method but should have been included. These qualities are closely related to the machine learning concepts of sensitivity and specificity. Challenges exist in achieving good coverage when there is an *imbalance* where the number of images containing POLs is much fewer than those without (i.e., there are a limited number of windmills in Iowa, so most locations do not contain a windmill).

The rest of this paper is organized as follows. First, we provide background on machine learning, as well as summarize relevant related work from the prior literature. Next, we introduce our approach for using machine learning to automate the construction of sampling frames for POLs, with the steps of the process illustrated through our windmill case study with its specific data and methods. Afterwards, we describe how we evaluate the sampling frames constructed using our approach, followed by an analysis of that evaluation within our case study. We conclude by discussing the implications of our results, the limitations of our study, and identifying critical areas of future work.

## Background and Related Work

Related to our study, prior literature has explored (a) the use of machine learning for automating sample frame construction in contexts that are different from

identifying POLs within aerial images, (b) using aerial images for sample frame construction using manual human labor (i.e., without machine learning), and (c) the use of machine learning to identify locations within aerial images outside of social science settings and sample frame construction. Together, these works inform how we might use machine learning to automate discovery of locations of POLs for sample frame construction using machine learning. We describe these related works below. To help contextualize this prior research, we first provide a general background on machine learning and information about the specific type of machine learning approach (convolutional neural networks) that we utilize for identifying POLs within images as part of our methodology (and in the cited related work).

## Machine Learning and Convolutional Neural Networks

Imagine that we want to accomplish a task that involves predicting labels for pieces of data; in this paper, that task involves predicting whether a given image contains a POL of interest (a positive predicted label of "yes") or not (a negative predicted label of "no").

The traditional programming approach to solve this problem would require a software programmer to implement a predetermined set of rules deciding when labels of "yes" and "no" should be predicted. For our problem, that requires domain expertise of all the possible ways a POL could appear in an image, which is often impossible to fully enumerate. Supervised machine learning, on the other hand, automates the process of determining how the computer should decide what label to predict for input data without requiring any domain expertise. A model is learned that finds patterns in example data provided by the researcher and then uses those patterns to predict a label for future data. In our case, a social scientist provides examples of images that contain the POL of interest, as well as examples of images that do not, so that the computer can find unique patterns present in both types of images. When non-example images are later given to the model, a correct label can be predicted if the model learned correct patterns from its examples.

What kinds of patterns the computer can learn and how it learns those patterns both depend on the specific type of machine learning model employed and the specific algorithm used to learn the model. With the goal of identifying POLs in aerial images, we utilize convolutional neural networks (CNNs), a form of deep learning most commonly applied to images. CNNs, illustrated generally in Figure 1, learn a hierarchy of information from the pixel data contained in images (i.e., the colors of each dot in both POLs and other entities in the aerial images). The first layer often learns how pixels form lines or contain relevant colors; the second layer often learns how lines organize into simple shapes or patterns of colors emerge; the third layer might learn how simple shapes and

color patterns form complex shapes (representing either POLs or their structural elements), and so on for all remaining layers until ultimately the final layer learns to identify whether an image contains a POL of interest or not.

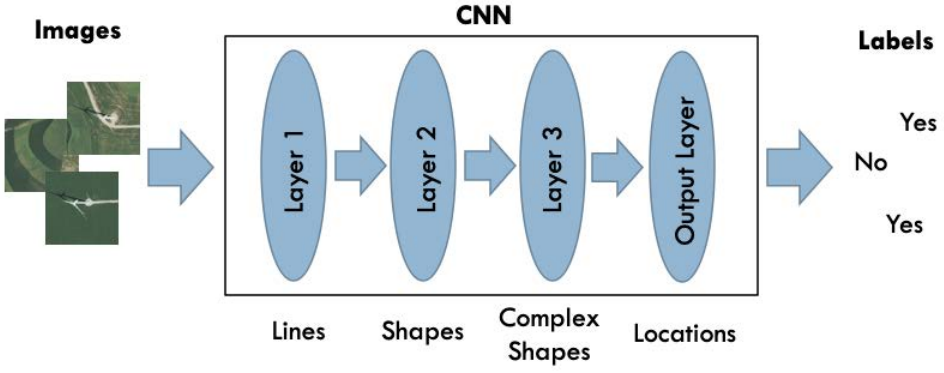



*Figure 1* Convolutional Neural Network Illustration

Different types of CNNs are capable of learning from different amounts of context provided about the content of images. One important type of CNN is *image classification* modeling, which learns to predict labels describing whether some type of content (here, POLs of interest, such as windmills) is present within an image. CNNs have achieved performance that exceeds the ability of humans to correctly label images with respect to a diverse range of content in the annual ImageNet competition (Russakovsky et al., 2015).

A second type of CNN that learns additional contextual information is *image segmentation* modeling, which outputs information describing *where* inside an image (i.e., which pixels/dots) the content of interest exists (e.g., which pixels represent a windmill in an aerial image), if it is present. For our purposes, image segmentation models are capable of highlighting *where* in the image a POL exists. If enough neighboring pixels are identified to belong to the POL of interest, then the entire image can be predicted as a "yes," and the physical location of the POL can be identified within the image.

For constructing sampling frames, image classification models have the advantage of being smaller models that often require less training and run faster with less computational requirements, whereas image segmentation models leverage more information during training to learn more informed models that might achieve higher performance. Our study experiments with both types in order to identify their strengths and weaknesses with respect to automating sample frame construction of POLs.

For further information about machine learning, including CNNs and how they might be applied to social science data, please refer to Goodfellow et al. (2016) and De Veaux and Eck (2021). The specific models of CNNs chosen for our study (ZFNet and VGG-16 as image classification models and U-Net as an image segmentation model) are described below in the subsection titled "Step 3: Train a CNN Model". More technical details about CNNs can also be found in Appendix A.

## Machine Learning for Sampling Frame Construction

Machine learning methods are beginning to be leveraged to refine and segment existing sampling frames or to construct new ones (Buskirk & Kirchner, 2020). Generally, unsupervised learning methods have been applied to take an existing population and partition it into subpopulations that serve as the basis of clusters or sampling strata. In this case, sampling units are already known in advance and are not discovered by the machine learning but are instead organized. For example, Burgette et al. (2018) compared three different unsupervised learning methods to create sampling strata in order to understand the range of care delivery structures and processes being deployed to influence the total costs of caring for patients over time. Buskirk et al. (2018a) applied k-means clustering to create geographically relevant sampling strata for a county-based RDD health survey using telephone bank information.

Additionally, supervised machine learning has been used to refine existing sampling frames. Garber (2009) used classification trees to predict eligibility of units included in a master mailing list for a survey targeting agricultural farms as part of a sampling frame refinement. Chew et al. (2018a) performed frame refinement by applying a two-category classification task to predict whether a satellite image scene is residential or non-residential within a gridded population sampling framework. Work is also emerging where supervised machine learning has been used to discover sampling units for a sampling frame, which we build on in this study. Chew et al. (2018b) also explored the use of object detection models such as Faster R-CNN (Ren et al., 2015) for enumerating buildings in satellite images in low- and middle-income countries.

The present study builds upon the prior use of machine learning to construct sampling frames by (a) considering a different type of sampling unit (POLs of interest), (b) learning from images, which present novel challenges for learning since they are *unstructured* (compared to the structured nature of many data sets represented as spreadsheets with curated columns of predictor variables), and (c) employing different types of CNNs, which are the state-of-the-art in the machine learning literature. Our study adds to the current literature by focusing on frame construction outright (rather than refining a sample frame or validating the eligibility of pre-selected sampled units; e.g., Chew et al., 2018a) and by

significantly extending prior work (Chew et al., 2018b) by considering a much larger study with several orders of magnitude more images, a larger geographic area, a broader range of metrics more closely related to sample frame quality, and using image segmentation for identifying more complex-shaped POLs at the *pixel*-level (instead of bounding boxes around buildings, which are typically rectangular shaped).

## Manual Sampling Frame Construction Using Aerial and Satellite Images

Aerial and satellite images have been used previously to *manually* create sampling frames (without machine learning). Some of the most common sampling frames constructed in this manner are of people or households either in remote areas or where traditional sampling frames (e.g., address-based sampling [ABS] or population census data) do not exist. For example, Himelein et al. (2014) used satellite images to sample random points and then canvassed a circle within a defined radius around those points to identify nomadic persons. Other researchers have sampled random points in satellite images to visually identify structures indicating the presence of people in areas including Malawi (Escamilla et al., 2014), Darfur (Lin & Kuwayama, 2016), Mozambique (Wagenaar et al., 2018), and Guatemala (Miller et al., 2020). Aside from sampling frames of people, the construction of sampling frames of agricultural areas has traditionally relied on manual review of aerial and satellite images, dating back at least as far as the work of Wigton and Borman (1978).

The present study builds upon the prior use of aerial and satellite images for sampling frame construction by automating the process to alleviate the time and monetary costs of hiring people to manually look carefully through images to identify POLs that should be added to a frame.

## Machine Learning with Satellite and Aerial Images

In the computer science literature, machine learning has been applied to aerial or related satellite images (jointly referred to as "remote sensing data") for identifying and counting various populations or structures. Naturinda et al. (2024) adapted deep learning-based algorithms to identify and count buildings from Unmanned Aerial Vehicle (UAV)-sensed images captured from a fairly suburban environment. Patel et al. (2022) also used UAV-sensed images to construct machine learning models that were able to detect and count the number of windows in a building. Hawkins et al. (2023) created machine learning models from drone images to automate the identification and counting of northern elephant seals in North America. For additional examples of other tasks being performed

with satellite images, please refer to Zhang et al. (2016), Zhu et al. (2017), and Song et al. (2019).

There is also a small collection of studies that have also used machine learning to identify images of windmills. After our study was conducted, Mridula and Sharma (2021) published work using another approach to image segmentation called Faster R-CNN (Ren et al., 2015) to identify windmills in satellite images for the purpose of identifying areas where windmills should be placed to optimize energy production. Stetco et al. (2019) reviewed the use of machine learning approaches other than CNNs to identify windmills in order to monitor the current condition of windmills in use for energy production. Both of these studies also find windmills as a location of interest, highlighting the appropriateness of this type of POL as a case study for our research, but neither studies how machine learning models can be used for the purpose of automating sampling frame construction.

Furthermore, Han et al. (2022) recently considered the use of image segmentation models (including U-Net, which we consider in our study) for the purpose of identifying buildings from remote sensing images (e.g., to aid in disaster recovery). Wagner et al. (2020) used U-Net machine learning models to identify buildings from satellite images in Brazil. McGlinchy et al. (2019) used U-Net models to learn to identify materials (e.g., concrete, asphalt) in urban areas, and Chen et al. (2021) learned models for identifying roads and sidewalks from remote sensing images. The use of U-Nets in these prior studies motivates its inclusion in our study, although our work differs in (a) automating the discovery of a different type of POL (windmills), and (b) we use our models for a very different purpose: constructing sampling frames that could be used to study the POLs identified (whereas the prior studies focused solely on the machine learning process, not its potential application).

## Constructing Sampling Frames of POLs Using Machine Learning

Our proposed approach for automatically constructing sampling frames of POLs using machine learning is detailed in Figure 2. In this section, we describe the steps involved in this process. For each step, we (a) detail the process required then (b) describe how the step was followed within our windmill sampling frame case study.

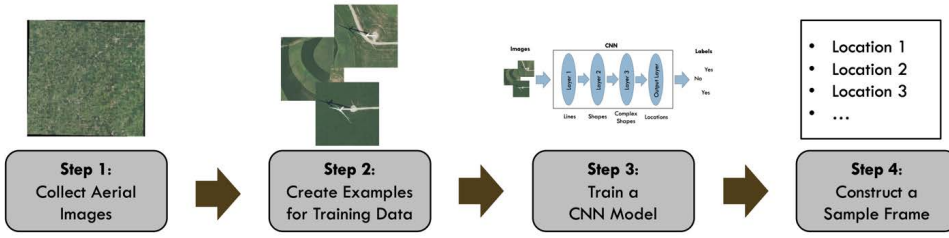



*Figure 2* Automated Sampling Frame Construction Process

## Step 1: Collect Aerial Images

**Process:** As a first step, images of the area of interest are collected by the researcher. Depending on the area of interest, this data could come from a variety of data sources. For instance, vendors such as Google (https://developers.google.com/maps) and Mapbox (https://docs.mapbox.com/help/dive-deeper/imagery/), as well as governmental agencies such as NASA (https://api.nasa.gov/) in the United States, offer satellite and aerial images (taken by satellites and airplanes, respectively) at various resolutions. Here, resolution refers to how much area on earth is covered by each image — analogously, the zoom level of the image. Higher resolution images are zoomed in closer to earth so that POLs are more visible. Appropriate resolution levels depend on the size of the POLs that the researcher is interested in, especially the size of the unique details of the POL of interest.

The specific images collected by the researcher should span two areas: (1) the *target application area* of interest, for which the location of POLs is unknown and a sampling frame is desired, and (2) a *training area*, for which the locations of some POLs are either known in advance or are manually discovered by people so that labeled training examples can be created (Step 2) and then provided to machine learning in order to train a model (Step 3) that will later be used to predict the locations of POLs in the target application area in order to build a sampling frame (Step 4). Once this process has been performed, the same machine learning models (learned in Step 3) could also be applied to find the same type of POL in new target application areas, requiring only the collection of aerial images of the new target application area of interest.

**Case Study Application:** For our case study, we collected aerial images covering the entire state of Iowa. The aerial images come from the USDA's NAIP[3], which is a public domain resource of freely offered aerial images taken by airplanes flying over the entire United States, updated every few years. Each

[3] We used the 2017 NAIP images for Iowa, which were the most recent available when the study began.

image in the NAIP covers an entire county within a state, so an entire state can be viewed by considering each of the individual counties within the state. For Iowa, the resolution of the images was 1 meter per pixel (i.e., each dot within the image records one square meter of area on the ground), so the original county-sized images were tens of thousands of pixels both tall and wide.

To illustrate what these aerial images look like, we provide examples containing and not containing a windmill in Figures 3a and 3d, respectively. Both of these images in the figure demonstrate the scale at which our models view aerial images to predict the presence of windmills.

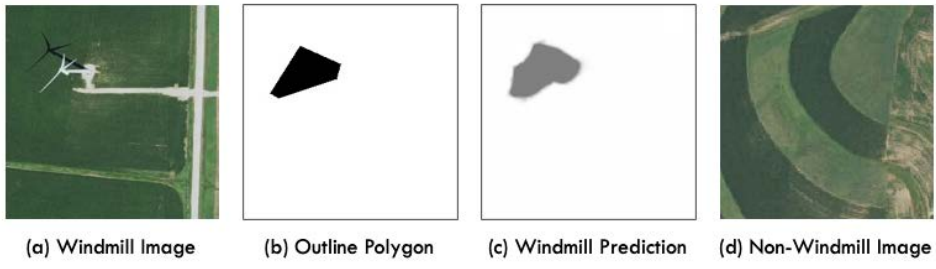



*Figure 3* Example Images of Windmill and Non-Windmill Locations

We used the ten counties in Iowa with the most windmills as our target application area[4], and the remaining 89 counties as our training area. This split of 10 vs. 89 counties was chosen because roughly 50% of the windmills occur in both categories (1,913 windmills in the target application area and 1,976 in the training area) so that we would have an almost equal distribution of windmills between the two areas. Using the densest counties for the target application area was the most efficient way to create training and evaluation data sets that had a sufficient number of windmills for modeling and evaluation without affecting the generalizability of our results.

## Step 2: Create Example Images for Training Data

**Process:** To train machine learning models that can predict whether images contain a POL of interest, the researcher must provide a set of labeled examples of both POL and non-POL locations from which the computer will learn. Step 2 of our process involves cropping (i.e., cutting out) examples of images that both contain and do not contain the POL of interest from the images collected in Step 1 above. Both types of examples are necessary so that the model can learn pat-

[4] The ten counties we used in our evaluation set are: Adair, Cass, Franklin, Hancock, Mitchell, O'Brien, Pocahontas, Story, Winnebago, and Worth.

terns that identify the key characteristics of the POL. For instance, by learning from images that do and do not contain windmills, the model can learn information such as most windmills have three large white blades centered on a tall white pole, whereas areas without windmills lack these visuals.

It is important to note that the number of examples needed for the training data depends on many factors. First, the more diversity that exists in the various ways the POL of interest might visually appear, the more examples are needed (e.g., windmills have a standard look, whereas playgrounds have different appearances in different regions). Second, the more similar the type of POL of interest is to other types of POLs not of interest to the researcher, the more examples are needed (e.g., windmills tend to have access roads that resemble residential streets and driveways). However, the number of examples needed for training should always be far fewer than the number of non-overlapping images that could be cropped from the target application area.

**Case Study Application:** In our case study, two undergraduate students used the open-source QGIS software (https://qgis.org) to load each of the counties' aerial images from the NAIP, found each windmill within the larger county-wide image, and then drew and saved polygons around each windmill so that square images containing the windmills could be cropped out of the county-wide image. For instance, the corresponding polygon for the windmill in Figure 3a is illustrated in Figure 3b. Each example image covered an area of 256 meters tall and wide (i.e., 256x256 pixels given the resolution of 1 meter per pixel), which is on par with the standard size[5] of images used with machine learning models and strikes a balance between the area covered by the aerial images and the appropriate visibility of windmills.

The students' search for windmills to create examples of our POL of interest in the training area was guided by the U.S. Wind Turbine Database[6]. This database contains information about every windmill in the United States, including the location of the windmill in GPS coordinates. We also used this database as our gold standard for evaluating the coverage of our sampling frames in Step 4. Part of the reason we chose windmills as our case study is the availability of this resource – it offers a frame for the total population, without which we would not be able to precisely measure over- and under-coverage when evaluating the sam-

[5] Most, if not all, CNN models, including ZFNet (Zeiler & Fergus, 2014), VGG-16 (Simonyan & Zisserman, 2015), and U-Net (Ronneberger et al., 2015), typically require relatively small inputs, such as 256x256 images, in order to create a model with a realistic number of parameters (e.g., neural network weights) that is tractable with modern computing hardware. Recall from the previous subsection titled "Step 1: Collect Aerial Images" that the entire county images in the NAIP are tens of thousands of pixels wide and tall, which are not feasible as inputs for CNN models.

[6] We used the first public release of the USWTDB from April 19, 2018. This was the closest in date to the 2017 NAIP images used in our study.

pling frames constructed with our machine learning models trained on aerial images.

To create examples of images that do not contain windmills, we randomly generated locations in the training area that are not within 256 meters of an identified windmill, then cropped square images of the same size as the windmill examples. By randomly sampling locations, we aimed to create a diverse set of non-windmill locations that describe a wide variety of other types of POLs (e.g., farms, parks, residential areas, commercial buildings, etc.). Other processes could also be used to generate images that do not contain the POL of interest, such as manually cropping images from the training area.

In total, our example data consisted of 1,976 labeled images that contained examples of windmill POLs and up to 17,784 labeled images that contained examples of non-windmills[7]. Although some time must be spent by the researcher (or hired for a monetary cost) to create and label the example images, it is still a small fraction of work that would instead be needed to manually inspect the entire target application area of interest, as we discuss in the Results and Discussion sections.

## Step 3: Train a CNN Model

**Process:** The third step of the process is to learn a machine learning model from the positive and negative example images created in Step 2. Because we are working with image data (in the form of aerial or satellite images), we propose using convolutional neural networks (CNNs, described in the first subsection of the Background and Related Work) as the machine learning model in our approach.

In most applications where a researcher desires a sampling frame of POLs, the POLs will be relatively rare within the area of interest (both the target application area and the training area). To illustrate, there were less than 4,000 windmills within the over 145,000 $km^2$ area of Iowa, which could be partitioned into over 2 million non-overlapping images of the dimensions appropriate for most CNN models (e.g., 256 x 256 pixels in our case study, given that images are available at the 1 meter per pixel resolution for Iowa in the NAIP). Thus, our entire data set would have less than 4,000 unique positive examples (windmill POLs) compared to over 2 million unique negative examples (non-windmill POLs). This represents a ratio of less than 1 positive example for every 500 negative examples (1:500).

[7] Our training area data included every example of a windmill in the 89 training area counties so that we could maximize the amount of information available about how windmills might appear in aerial images to our machine learning models. We describe in the following subsection how we selected the number of non-windmill examples to include in the training area data in order to maximize the quality of learning given the natural rarity of POLs of interest within the total training area.

Unfortunately, supervised machine learning (such as training CNNs) is very sensitive to the ratio of positive and negative examples in the training data—if there are too many negatives for every positive example, then any model will almost certainly learn to always make a negative prediction (i.e., not a POL of interest) with virtually 100% accuracy[8]. As a result, the models would fail to identify *any* POLs of interest and thus always end up with an empty sampling frame! This is a problem commonly referred to in the machine learning literature as the class imbalance problem.

One of the simplest and most robust solutions to addressing class imbalance is to undersample the majority negative examples. In this process, every positive example (with a POL) is retained in the training data used to learn the model, but only a randomly sampled subset of the more common negative examples is retained (and the others discarded). For our study, we utilized uniform random sampling to choose the negative (non-POL) examples to retain, although we hypothesize that more elaborate sampling schemes (e.g., cluster sampling where non-POL images are clustered based on the type of location they represent, such as farms, households, businesses, highways, etc.) might further improve performance, which we leave as future work.

The quantity of majority negative examples to retain is a key hyperparameter that affects the quality of learning – retaining too many fails to mitigate the class imbalance problem, while retaining too few removes too much of the diversity of negative examples, reducing the ability of the machine learning models to identify patterns that are representative of negative examples (e.g., key characteristics of non-windmill POLs such as residential areas, commercial buildings, and rural land). Thus, a researcher might need to fine-tune this hyperparameter, which we performed as a pre-experiment described in detail in Appendix B. Ultimately, we found that ratios between 1:1 and 1:9 (positive:negative) POL examples in the training set were ideal for our study, as described below.

**Case Study Application:** We consider two popular image classification CNNs called ZFNet (Zeiler & Fergus, 2014) and VGG-16 (Simonyan & Zisserman, 2015) that have been demonstrated previously to achieve great success without requiring the significant computational complexity[9] of some of the most complicated approaches such as Inception (Szegedy et al., 2015).

We also consider the popular U-Net approach (Ronneberger et al., 2015) to learn image segmentation models. The polygons created by the undergraduate students also served a secondary purpose in providing the context of exactly

[8] For instance, the accuracy of all negative predictions for a ratio of 1:500 positive:negative data would be 99.8%.

[9] Our models were learned on computers using only a single GPU (a special card that handles graphics and can be used to speed up deep machine learning), whereas the most complicated models require learning across multiple GPUs and often multiple computers in a cluster infrastructure, making them incompatible with the computing infrastructure available to many social science researchers.

where in the cropped images each windmill exists. Indeed, the pixels predicted by a trained U-Net model as belonging to a windmill for the aerial image given in Figure 3a are provided in Figure 3c, which is notably *very consistent* with the polygon drawn by the undergraduate student for that windmill in Figure 3b.

We implemented our CNN models using the industry-leading TensorFlow library for deep learning (Abadi et al., 2016) in the Python programming language. Our source code is publicly available on GitHub[10]. The ZFNet, VGG-16, and U-Net approaches followed the sets of layers originally proposed in their original citations, with the exception that the output layers were modified to only output binary predictions either at the image level (for the image classification models learned using ZFNet and VGG-16) or at the pixel level (for the image segmentation models learned using U-Net). This was necessary since the approaches were originally proposed to learn to predict more classes of labels (i.e., multiple types of image content), whereas we only require a "yes" or "no" prediction in order to identify POLs of interest and construct sampling frames. We chose to train every model from scratch since pre-trained models for image segmentation using U-Net are not as common as image classification with ZFNet and VGG-16, and it would not be a fair comparison to use pre-trained models for image classification and not image segmentation.

For reproducibility of our evaluation, we provide the key hyperparameters describing how the models were trained in Appendix C, along with a description of a pre-experiment conducted to decide how to tune those hyperparameters (a process commonly used to learn the best possible machine learning models), including the undersampling ratio to address class imbalance, in Appendix B. Different machine learning approaches respond differently to the undersampling ratio, so we explored several values. We ultimately found that the ideal ratio was 1:1 (1,976 windmill and non-windmill examples) for UNet and 1:9 for both ZFNet and VGG-16 (1,976 windmill and 17,784 non-windmill examples).

## Step 4: Construct a Sampling Frame

**Process:** Once a machine learning model has been trained on example images, it can be used to predict whether images from the target application area contain the POL of interest (e.g., windmills) or not. The positive predictions then represent locations where a POL of interest exists, which can be aggregated to form a sampling frame.

First, the aerial or satellite images collected in Step 1 for the target application area must be processed so that they can be fed into the machine learning model trained in Step 3 in order to predict the locations of the POL of interest for the sampling frame. Recall that CNN models typically take as input small

[10] https://github.com/OberlinAI/EyeSpyAPSU

images (with widths and heights in the hundreds of pixels, e.g., 256 x 256 pixels), whereas the images that span a large geographic area are often much, much larger (with widths in the tens of thousands of pixels for counties or more for larger areas, depending on the underlying geography).

We propose partitioning the entire target application area of interest into contiguous, non-overlapping smaller areas and creating images for each of the smaller areas. Think of this as overlaying a grid on top of the target area of interest, then making predictions for each location within the grid. This process can be automated by software to create the thousands or millions of images needed to span the entire target application area of interest, avoiding the manual effort needed in Step 2 to create the smaller number of example images from the training area.

Next, each smaller image from the target application area is given to the machine learning model, and a prediction is made whether or not that image (and its underlying smaller area) contains a POL of interest. If the model predicts "yes", then a representative location of the smaller area (e.g., the GPS coordinates of its center) is added to the sampling frame. By making predictions for every image of every smaller area spanning the entire target application area, the POLs of interest within the target application area will hopefully be identified.

**Case Study Application:** For the target application area, we followed the process described above. In particular, we downloaded all aerial images for the 10 evaluation counties[11]. Next, we overlayed a 256-meter by 256-meter grid on top of each county (matching the same size used in the training example images since CNN models require consistently sized input images). We then spliced the 256-meter by 256-meter areas divided by the grids of each county in order to create a collection (i.e., testing set) of contiguous, non-overlapping 256 x 256-pixel images covering the entire evaluation counties. Every location within the evaluation counties was then contained in exactly one of these smaller images that could be evaluated by our CNN models to identify windmill POLs. As described in Table 1 comparing our training and target application areas, this process resulted in a total of 304,941 images (1,913 containing windmills and 303,028 without windmills) for the target application area (i.e., our testing set). We fed each of those images into our CNN models and saved the locations of positive predictions as a separate sampling frame for each model.

[11] The 10 counties we used in our evaluation set (i.e., test set) are: Adair, Cass, Franklin, Hancock, Mitchell, O'Brien, Pocahontas, Story, Winnebago, and Worth.

*Table 1* Counts of Images in Training and Target Application Area Data

| | Windmill Images | Non-Windmill Images |
|---|---|---|
| Training Area (89 counties) | 1,976 | UNet: 1,976<br>ZFNet & VGG-16: 17,784 |
| Target Application Area (10 counties) | 1,913 | 303,028 |

## Sampling Frame Evaluation

Provided that the machine learning model makes accurate predictions, the sampling frame will have good coverage of the POLs in the area studied by the researcher. For evaluation purposes only, we can determine the correctness of predictions if we know the locations of the POLs of interest in the target application area in advance. For the "ground truth" in our case study, each partitioned image in the target application area was considered to possess a windmill (and have an actual positive label) if the center of a windmill was contained in the image; else, the image was not considered to possess a windmill (and have an actual negative label).

We evaluate the performance of our approach in the target application area of our windmill POL case study from two different perspectives. From the sampling frame perspective (which is our ultimate task), we evaluate the coverage of the sampling frames. We calculate counts of (a) the number of windmills in the target application area correctly predicted by the models and added to the resulting sampling frame (i.e., true positive predictions, TP), (b) the number of actual windmills mispredicted by the models (i.e., false negative predictions, FN), representing under-coverage in the sampling frames, and (c) the number of non-windmill locations mispredicted by the models (i.e., false positive predictions, FP), representing over-coverage in the sampling frames.

From the machine learning perspective, we calculate several performance measures related to the sampling frame measures and commonly used in the social science literature. Sensitivity (also referred to as "recall" or "true positive rate") measures the proportion of images with windmills correctly predicted, indicating how well the models learned to find patterns related to the POL of interest. Specificity (also referred to as true negative rate) measures the proportion of images without windmills that were correctly predicted, indicating how well the models learned to find patterns covering the diversity of other types of POLs (so that they can be excluded from the sampling frame). Balanced accuracy is the average of sensitivity and specificity, measuring how well the model learns to accurately predict both the POLs we are interested in and those we are not,

which is a holistic perspective of overall performance given the class imbalance in the data set. Common to other machine learning studies, we also report overall accuracy, which measures the total proportion of images correctly labeled (both windmill and non-windmill images), as well as precision, which measures the proportion of images correctly labelled among those predicted by the model to have a windmill (this metric is also known as positive predictive value, PPV).

Because we relied on undersampling to address the class imbalance problem (cf. subsection "Step 3: Train a CNN Model"), the random sampling of non-windmill locations introduces a confounding factor that could affect the quality of machine learning. To reduce the variance introduced, our analysis is based on repeating our proposed approach 30 times using different random seeds to produce 30 different sets of undersampled training data sets and thus 30 models of each type of CNN. The results presented below represent the average of the performances of the 30 sampling frames for each type of CNN.

## Results

In this section, we present the results of our evaluation of the quality of sampling frames of windmill locations constructed with our machine learning models. We first focus on the sampling frames constructed by the image classification models learned using ZFNet and VGG-16, followed by the sampling frames constructed by the image segmentation models learned using U-Net. We present the results of all three approaches in Table 2.

### Image Classification Results

From Table 2, we make several key observations about the quality of the sampling frames of windmill locations constructed using the ZFNet and VGG-16 image classification models. First, the models learned by both CNN approaches resulted in sampling frames that identified the majority of desired POLs for inclusion: at least 1,078 of the 1,903 windmills in the ten evaluation counties were correctly identified (with sensitivity rates of 56.36% and 57.62%, respectively).

The very large class imbalance in the data (recall that there were 158.4 times as many non-windmill locations as windmill locations in the evaluation counties; cf., Table 1) implies that such high sensitivity rates are a noteworthy achievement, since a model that only randomly guesses the presence of a POL (i.e., the null hypothesis, based on the observed distribution of windmill locations) would only achieve a sensitivity of 0.63% (1/158.4). This indicates that our undersampling approach and hyperparameter tuning during training (cf., Appendices B and C) led to the machine learning finding visual patterns that are indeed related to the presence of a POL in this complicated task. We expect that

for other applications where the POL of interest is less under-represented in the area of interest, even better sampling frames could be constructed with image classification models.

*Table 2* Sampling Frame and Machine Learning Performances

| Performance Measure | ZFNet | VGG-16 | U-Net(400) | U-Net(0) |
|---|---|---|---|---|
| Windmills Identified (TP) | 1,078 | 1,102 | 1,521 | 1,656 |
| Under-Coverage (FN) | 835 | 811 | 392 | 257 |
| Over-Coverage (FP) | 4,015 | 3,766 | 3,073 | 5,544 |
| Non-Windmills Identified (TN) | 299,013 | 299,262 | 299,955 | 297,484 |
| Sensitivity (Recall) | 56.36% | 57.62% | 79.53% | 86.56% |
| Specificity | 98.68% | 98.76% | 98.99% | 98.17% |
| Balanced Accuracy | 77.52% | 78.19% | 89.26% | 92.37% |
| Accuracy | 98.41% | 98.50% | 98.86% | 98.10% |
| Precision (PPV) | 27.52% | 26.42% | 36.18% | 25.75% |

However, the sampling frames constructed with both ZFNet and VGG-16 also suffer from problems in their coverage. Both approaches mispredicted at least 811 of the actual windmill locations, causing the sampling frames to suffer from under-coverage. Additionally, both approaches also mispredicted at least 3,766 actual non-windmill locations, causing more actual non-windmill locations to be added to the sampling frame than actual windmill locations. Thus, neither CNN led to learning that can fully automate the construction of sampling frames. However, they could provide an important filtering step that greatly reduces the amount of manual effort needed to finalize a good sampling frame, as we propose below in our Discussion section.

## Image Segmentation Results

We also make several important observations about the quality of the sampling frames of windmill locations constructed by the U-Net image segmentation models. In Table 2, we present two sets of results for U-Net: results where an image is predicted as having a windmill if more than 400 of the pixels in the image are predicted to belong to a windmill, which we refer to as U-Net(400), and additional results where an image is predicted as having a windmill if more than 0 pixels (i.e., *any* pixel) is predicted to belong to a windmill, referred to as U-Net(0). These represent different thresholds for transforming contextual knowledge about the locations of windmills within images into predictions of whether a windmill exists at all. We initially chose 400 pixels as a threshold because that is close to the sizes of the windmill polygons labeled by the under-

graduate students in the data sets, but we also consider 0 pixels as another baseline since it indicates whether any part of a windmill was found.

Considering the results in the U-Net(400) column of Table 2, we observe that the models learned using U-Net resulted in *better performance* than the two image classification models: U-Net correctly identified many more windmill locations (1,521 vs. 1,102), as well as correctly identified more non-windmill locations (299,955 vs. 299,262), altogether achieving a 38% and 14% improvement in sensitivity and balanced accuracy, respectively, over the VGG-16 models. Thus, the additional context learned by image segmentation models produced more useful patterns from the example data since they were guided by the polygons labeling which pixels belonged to the windmills.

As a result of higher sensitivity and specificity, we also see that the sampling frames constructed using U-Net(400) achieved less under-coverage and less over-coverage than the image classification models analyzed in the previous subsection: only 392 windmills were missed by U-Net(400), compared to 811 by VGG-16, and only 3,073 non-windmill locations were erroneously added to the sampling frame by U-Net(400), compared to 3,766 by VGG-16. Therefore, the sampling frames constructed using U-Net(400) are of higher quality than the sampling frames constructed by either image classification model. Indeed, paired t-tests with Bonferroni correction indicated statistically significant improvements in performance[12] for U-Net on both windmills identified (TP) and under-coverage (FN) at the $\alpha = 0.01$ significance level (highest $p < 0.001$ and lowest magnitude $t = 18.874$).

Finally, we compare the results of U-Net(0) with U-Net(400), presented in Table 2, where the former predicts an image as having a windmill if any pixel (instead of more than 400 pixels) is predicted to belong to a windmill. We observe that this type of model achieved even higher sensitivity (86.56% vs. 79.53%), resulting in less undercoverage (only 257 missed windmills out of 1,903 locations) than the well-performing U-Net(400). The tradeoff is that U-Net(0) also achieved worse over-coverage than U-Net(400), as it mispredicted 5,544 locations as a windmill when the location does not in fact contain a windmill. Using the power of hindsight, we would suggest using 0 pixels as the threshold for predicting an image as containing a windmill since under-coverage cannot be improved without manually reviewing *all* images, whereas the small amount of additional over-coverage can be eliminated with a little more manual image review (only an extra 2,606 images), as we discuss in the next section.

[12] All three methods achieved statistically equivalent performance on over-coverage (FP). Wilks-Shapiro tests were used to determine that TP and FN counts followed a normal distribution ($p > 0.05$ and $W > 0.9735$), so we used paired t-tests instead of paired Mann-Whitney U-tests.

# Discussion

## Implications

From the analysis of our results above, we can now answer our two research questions and evaluate our two hypotheses from Section 1. First, we find that both image classification and segmentation models were able to correctly discover the majority of POLs of interest within the area studied in the case study. This *confirms our first hypothesis* that machine learning can indeed be used to identify sampling units for a sampling frame of POLs, although we find that the performance is strongly dependent on the type of model used. As a result, our machine learning approach for automatically constructing sampling frames could be a valuable resource for social scientists studying populations of POLs, or for whom identifying POLs could be valuable for developing related sampling frames of people (e.g., clean energy producers, families using public playgrounds).

Furthermore, we found that the image segmentation models learned by U-Net resulted in statistically significantly better sampling frames. The frames constructed using U-Net contained a higher proportion of correctly identified sampling units (79.53% vs. 57.62%) and, consequently, significantly less undercoverage. This *confirms our second hypothesis* that image segmentation models successfully utilized their additional context learned to construct sampling frames with better coverage. As a result, we suggest employing image segmentation models to construct sampling frames of POLs whenever the researcher can afford the added time required to label *where* the POL exists within the example images. This matches the recent trends in applying machine learning to remote sensing data for other tasks described in subsection "Machine Learning with Satellite and Aerial Images".

At the same time, the sampling frames created by all three approaches (ZFNet, VGG-16, and U-Net) unfortunately suffered from some amount of over- and under-coverage to different extents, so none of the models performed perfectly. As a result, there remains room for improvement in our method.

To overcome the over-coverage issues introduced by our approach, we recommend that a person manually check the images predicted to have a POL by the machine learning model and then apply a final manual decision of whether an identified POL should be added to a sampling frame. In this case, the machine learning model operates as a filtering process to aid social science researchers. Although this adds some manual effort by a researcher, the number of images they need to consider is *drastically* reduced from the total number of images covering the area of interest. Indeed, only 5,093; 4,868; and 4,594 images were predicted as having a windmill by the ZFNet, VGG-16, and U-Net models, respectively. If each "yes" prediction were manually reviewed to remove all over-coverage, that would still result in less manual labor needed to refine the

sampling frame (i.e., at least 98.33% of the 304,941 images comprising the entire target application area in our case study could be marked for exclusion by our approach).

Under-coverage in the sampling frames is a little bit more difficult to eliminate, as it requires understanding why some POLs of interest were missed by the machine learning models. Manual review of the images for the non-windmill predictions would not be helpful in this case, due to the vast imbalance that causes many more images to naturally lack the POL of interest. As a first step, we suggest the following refinement to the approach, which could be studied as future work. All neural networks, including CNNs, determine a probability that each label should be predicted for the input image. For binary predictions (such as whether or not a POL exists), a threshold[13] determines whether a positive (e.g., windmill POL) or negative (e.g., non-windmill POL) prediction should be made. Instead, the images that received negative predictions but were closest to the predictive threshold could be filtered and a small subset manually reviewed. Notably, these represent the images for which the models were most uncertain when assigning labels and therefore could be the images most likely to contain the POL of interest but also be mispredicted by the models. This would result in a similar review process to our proposed solution for over-coverage with low manual effort required.

We manually reviewed some of the mispredictions made by our models, and we discovered an explanation for some of the over- and under-coverage. Recall (cf., Section "Step 4: Construct a Sampling Frame") that our target application area used for evaluation spliced the total land mass of 10 counties into smaller images (representing a 256-meter by 256-meter area) that could be fed into our CNN models. This splicing was agnostic to the true locations of POLs, so occasionally a POL naturally fell near the boundary between two images (or four images if it was near a corner), illustrated in Figure 4. As a result, part of the image of the POL would be in two (or four) images. When large enough parts of the same windmill occurred in multiple images (e.g., Figures 4a and 4b), the U-Net image segmentation models identified pixels as belonging to what was the same windmill in all of those images, leading to false positive predictions and over-coverage (since only one image was considered to contain the windmill as ground truth). On the other hand, when only a small incomplete part of a windmill was in the "ground truth" image (e.g., Figure 4c), the U-Net models might not have seen enough of the windmill to convince themselves that any pixel belonged to a windmill, leading to many of our false negative predictions and under-coverage. As a refinement for our method in Step 4 when using image

[13] Given the different under-sampling ratios for the different types of models, we used different probability thresholds from the logistic regressor in the output layers to determine when a label of "yes" should be predicted for an input image: 10% for ZFNet and VGG-16 and 50% for U-Net, each equal to 1 / (ratio + 1).

segmentation models, we could instead compare which pixels are identified as belonging to a POL *across* images, then (a) treat neighboring areas as a single POL to reduce false positives and over-coverage, and (b) allow slight deviations between the center of the identified pixels and true POL locations to reduce false negatives and under-coverage (at the cost of a small amount of location precision).

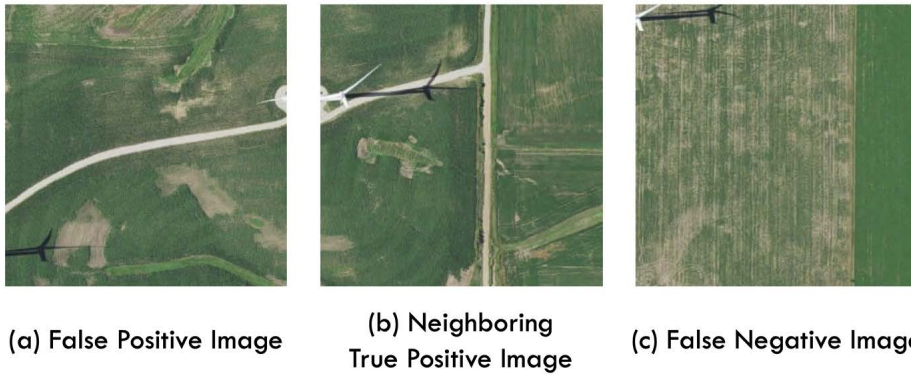



*Figure 4* Example Mispredictions Due To Large Geographic Area Splicing

## Limitations

Although the results of our case study have demonstrated many successes of our proposed approach, it is important to also note several limitations of the study. First, we only considered a single type of POL, and the results might not generalize to all possible POLs in which social scientists are interested. Indeed, windmills have very distinct characteristics (three blades and a vertical pole with a homogeneous white color that also produce a distinctive shadow) that might be easier to visually recognize than other POLs of interest, although they are complicated by being irregularly shaped (depending on the perspective of the satellite/aerial images). We also only considered a single area of interest – the ten counties in the state of Iowa with the largest number of windmills. Our result also might not generalize to all other areas of interest, even for the same type of POL.

However, given that CNN models have been demonstrated to achieve superior labeling ability to humans on a diverse set of image content (Russakovsky et al., 2015), we expect the method will generalize well to many types of POLs of interest. So long as (a) the POLs have distinctive visual characteristics that differentiate them from other types of POLs (e.g., jungle gyms and swing sets in playgrounds or unique architectural details for places of worship), and (b) the satellite or aerial images are available with sufficient resolution that the distinctive characteristics are visible.

Another limitation of our study is that we started with a gold-standard frame of windmill locations in the U.S. Wind Turbine Database, which obviously will not be the case for real-world applications (else there would be no need to construct a sampling frame in the first place). On the one hand, our machine learning models did not exploit this known frame information, but instead the frame was used to evaluate the sampling frame created by our machine learning models. So, our methodology still generalizes when a frame is unknown *a priori*, as our case study was intended to simulate. On the other hand, the known frame did guide the development of our training area images – we knew in advance where to look to find example images of windmill and non-windmill locations in the aerial images from the NAIP. In practice, a researcher would instead often start with a partial frame that provides some example locations of POLs, from which a training set could be constructed (e.g., knowing the locations of some windmills in a different area, as in our 89 counties used for the training set). Or in the worst case, the researcher would have to manually search through a subset of the satellite/aerial images for some examples of POL and non-POL locations, which adds some manual labor compared to our process but is still less than manually reviewing the entire area of interest (i.e., the more than 300,000 images in our evaluation set).

Finally, there was about a year gap between when the NAIP images in our data set were collected by the USDA (2017) and when the first version of the U.S. Wind Turbine Database was released (April 2018). Thus, it is possible that some windmills were constructed between when the images were collected and when the gold-standard frame was created that we used to evaluate our models. Indeed, the USWTDB identifies 197 windmills that were operationalized in 2017[14], some of which could have been constructed after the NAIP images were taken. This could have resulted in both the mislabeling of some of the training set images (where 188 windmills were operationalized in 2017 in our 89 training counties) or false negative predictions in our evaluation set (where nine windmills were operationalized in 2017 in our ten evaluation counties). Although these were the best data sets available when our study began, a closer matching in time between the images and ground truth information could have further improved the performance of our methodology, which might be less of a challenge in other applications of POL sampling frame construction.

## Future Work

While we applied our methodology to creating sampling frames of windmills, we foresee additional applications of it being used to create sampling frames of

[14] Timing of when windmills became operational is only available at the year granularity and not month or day, so it is unknown when exactly the 2017 windmills would have been visible in aerial images.

other types of general POLs that are identifiable from satellite/aerial images, such as playgrounds or pickleball courts or loading docks at warehouses, for example. At the same time, it remains an open question how well our methodology would apply to creating sampling frames of *specific subsets* of related POLs with special characteristics such as playgrounds with and without awnings or pickleball courts with and without lights. Certainly, these special characteristics would need to be visually distinctive in the accompanying satellite/aerial images. Special attention may need to be paid to ensure that there are enough examples of the POLs with and without the special feature to further distinguish the specific subgroups of interest, and model customization might also be necessary to maximize differentiation of these subgroups.

Besides generalizing the application areas for which our method could apply, we also foresee opportunities in refining the specific machine learning processes that are deployed as part of our overall method to even more accurately discover POLs and produce more complete sampling frames. For example, in our current approach we only considered a single type of image segmentation model (U-Net) in our study, whereas others have been recently proposed (e.g., Mask R-CNN (He et al., 2017)) that might offer improvements in the patterns learned by the models.

We also only considered simple random samples of "no" examples when curating our under-samples during training to address class imbalance, whereas stratified sampling of different types of "no" locations could have produced more informed models. For example, given the vast farmland in Iowa, some types of non-windmill locations such as residential areas that have streets resembling the access roads leading to windmills might have been under-represented in the training examples, leading the models to overinflate the importance of roads in identifying windmills, increasing our false positive rates.

Finally, we also trained our models from scratch, whereas pre-trained models of ZFNet and VGG-16 from other image classification tasks can be downloaded and then retrained for new tasks. This process, called transfer learning, enables models to start with pre-trained general visual knowledge that is improved with new task-specific patterns. As social scientists begin to explore more applications of the approach we outline here and construct models predicting various types of POLs and post these results in the public domain, the opportunities to apply transfer learning methods will vastly improve. We look forward to seeing these models emerge over time!

## References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., … Zheng, X. (2015). *Tensorflow: Large-scale machine learning on heterogenous systems*. arXiv. https://doi.org/10.48550/arXiv.1603.04467

Adamu, C. D. (2025). Public primary school teachers' perception towards child abuse reporting practices in Nigeria. Primaryedu: *Journal of Elementary Education, 9*(1), 25-37. https://doi.org/10.22460/pej.v9i1.5475

American Wind Energy Association. (March 6, 2017). *U.S. wind generation reached 5.5% of the grid in 2016*. https://www.awea.org/MediaCenter/pressrelease.aspx?ItemNumber=9999

Brummet, Q., Masterton, M., & Smith, D. (2014). *Evaluation of commercial school and teacher lists to enhance survey frames (No. 2014-07)*. Center for Economic Studies, US Census Bureau. Retrieved May 2, 2025, from https://www.census.gov/library/working-papers/2014/adrm/carra-wp-2014-07.html

Burgette, L. F., Escarce, J. J., Paddock, S. M., Ridgely, M. S., Wilder, W. G., Yanagihara, D., & Damberg, C. L. (2019). Sample selection in the face of design constraints: Use of clustering to define sample strata for qualitative research. *Health Services Research, 54*(2), 509-517. https://doi.org/10.1111/1475-6773.13100

Buskirk, T. D., Bear, T., & Bareham, J. (2018a, October). *Machine made sampling designs: Applying machine learning methods for generating stratified sampling designs* [Conference presentation]. BigSurv18 Conference, Barcelona. Spain. Retrieved June 20, 2019, from https://www.bigsurv18.org/conf18/uploads/177/203/BuskirkBearBarehamBigSurv18ProgramPostedVersionFinal.pdf

Buskirk, T. D., & Kirchner, A. (2020). Why machines matter for survey and social science researchers: Exploring applications of machine learning methods for design, data collection, and analysis. In C.A. Hill, P.P. Biemer, T.D. Buskirk, L. Japec, A. Kirchner, S. Kolenikov and L.E. Lyberg (Eds.), *Big Data Meets Survey Science: A Collection of Innovative Methods* (pp. 9-62). Wiley. https://doi.org/10.1002/9781118976357.ch1

Buskirk, T. D., Kirchner, A., Eck, A., & Signorino, C. S. (2018b). An introduction to machine learning for survey researchers. *Survey Practice, 11*(1). https://doi.org/10.29115/SP-2018-0004

Chabot, D., Dillon, C., & Francis, C. M. (2018). An approach for using off-the-shelf object-based image analysis software to detect and count birds in large volumes of aerial imagery. *Avian Conservation and Ecology, 13*(1), Article 15. https://doi.org/10.5751/ACE-01205-130115

Chen, Z., Luo, R., Li, J., Du, J., & Wang, C. (2021). U-Net based road area guidance for crosswalks detection from remote sensing images. *Canadian Journal of Remote Sensing, 47*(1), 83-89. https://doi.org/10.1080/07038992.2021.1894915

Chew, R. F., Amer, S., Jones, K., Unangst, J., Cajka, J., Allpress, J., & Bruhn, M. (2018a). Residential scene classification for gridded population sampling in developing countries using deep convolutional neural networks on satellite imagery. *International Journal of Health Geographics, 17*(1), Article 12. https://doi.org/10.1186/s12942-018-0132-1

Chew, R., Jones, K., Unangst, J., Cajka, J., Allpress, J., Amer, S., & Krotki, K. (2018b). Toward model-generated household listing in low- and middle-income countries using deep learning. *ISPRS International Journal of Geo-Information, 7*(11), Article 448. https://doi.org/10.3390/ijgi7110448

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., & Fei-Fei, L. (2009). ImageNet: A large-scale hierarchical image database. In *Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition* (pp. 248-255). IEEE. https://doi.org/10.1109/CVPR.2009.5206848

De Veaux, R. D. & Eck, A. (2021). Machine learning methods for computational social science. In U. Engel, A. Quan-Haase, S.X. Liu, & L. Lyberg (Eds.), *Handbook of Computational Social Science, Vol. 2: Data Science, Statistical Modeling, and Machine Learning Methods* (pp. 291-321). Routledge. https://doi.org/10.4324/9781003025245-21

U.S. Energy Information Administration (EIA). (2019). *How we chose buildings for the 2018 CBECS*. Retrieved on May 8, 2022, from https://www.eia.gov/consumption/commercial/reports/2018/methodology/sampling.php

Escamilla, V., Emch, M., Dandalo, L., Miller, W. C., Martinson, F., & Hoffman, I. (2014). Sampling at community level by using satellite imagery and geographical analysis. *Bulletin of the World Health Organization*, *92*(9), 690–694. https://doi.org/10.2471/BLT.14.140756

Garber, S. C. (2009). Census mail list trimming using SAS data mining. In *Department of Agriculture, National Agricultural Statistics Service, RDD Report 09-02*.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.

Han, Q., Yin, Q., Zheng, X., & Chen, Z. (2022). Remote sensing image building detection method based on Mask R-CNN. *Complex and Intelligent Systems, 8*, 1847-1855. https://doi.org/10.1007/s40747-021-00322-z

Hawkins, S., Lundberg, A., Lundberg, A., Oliver, G., & Condit, R. (2023). Estimating a seal population using automated counts of drone photographs. In *Proceedings of the 2023 SPIE Optical Engineering + Applications Conference, 12675*. SPIE. https://doi.org/10.1117/12.2676277

He, K., Gkioxari, G., Dollar, P., & Girshick, R. (2017). Mask R-CNN. In *Proceedings of the 2017 IEEE International Conference on Computer Vision* (pp. 2980–2988). IEEE. https://doi.org/10.1109/ICCV.2017.322

Himelein, K., Eckman, S., & Murray, S. (2014). Sampling nomads: A new technique for remote, hard-to-reach, and mobile populations. *Journal of Official Statistics*, *30*(2), 191-213. https://doi.org/10.2478/jos-2014-0013

Kingma, D. P. & Ba, J. L. (2015). Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference on Learning Representations*. https://doi.org/10.48550/arXiv.1412.6980

Lin, Y. & Kuwayama, D. P. (2016). Using satellite imagery and GPS technology to create random sampling frames in high risk environments. *International Journal of Surgery*, *32*, 123-128. https://doi.org/10.1016/j.ijsu.2016.06.044

McGlinchy, J., Johnson, B., Muller, B., Joseph, M., & Diaz, J. (2019). Application of UNet fully convolutional neural network to impervious surface segmentation in urban environment from high resolution satellite imagery. *In Proceedings of the 2019 IEEE International Geoscience and Remote Sensing Symposium* (pp. 3915-3918). IEEE. https://doi.org/10.1109/IGARSS.2019.8900453

McMahon, C. R., Howe, H., van den Hoff, J., Alderman, R., Brolsma, H., & Hindell, M. A. (2014). Satellites, the all-seeing eyes in the sky: Counting elephant seals from space. *PLOS ONE*, *9*(3), 1-5. https://doi.org/10.1371/JOURNAL.PONE.0092613

Miller, A. C., Rohloff, P., Blake, A., Dhaenens, E., Shaw, L., Tuiz, E., Grandesso, F., Mendoza Montano, C., & Thomson, D. R. (2020). Feasibility of satellite image and GIS sampling for population representative surveys: A case study from rural guatemala. *International Journal of Health Geographics*, *19*. Article 56. https://doi.org/10.1186/s12942-020-00250-0

Mridula, A. & Sharma, S. (2021). WMDeepConvNets: Windmill detection using deep learning from satellite images. In: K. Kotecha, V. Piuri, H. Shah, R. Patel (Eds). *Data Science and Intelligent Applications. Lecture Notes on Data Engineering and Communications Technologies, 52*. Springer. https://doi.org/10.1007/978-981-15-4474-3_19

Naturinda, E., Omia, E., Kemigyisha, F., Aboth, J., Kabenge, I., & Gidudu, A. (2024). Counting buildings from unmanned aerial vehicle images using a deep learning based approach. *South African Journal of Geomatics, 13*(1), 83-93.

Nelson, D., Kreps, G., Hesse, B., Croyle, R., Willis, G., Arora, N., Rimer, B. K., Viswanath, K. V., Weinstein, N., & Alden, S. (2004). The health information national trends survey (HINTS): development, design, and dissemination. *Journal of Health Communication, 9*(5), 443-460. https://doi.org/10.1080/10810730490504233

Patel, D., Chepuri, S., Thakur, S., Harikumar, K., Sarvadevabhatla, R. K., & Krishna, K. M. (2023). *Automated Detection and Counting of Windows using UAV Imagery based Remote Sensing*. arXiv. https://doi.org/10.48550/arxiv.2311.14635

Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster R-CNN: Towards real-time object detection with region proposal networks. In C. Cortes, N. Lawrence, and D. Lee (Eds). *Advances in Neural Information Processing Systems, 28*. Curran Associates, Inc. https://doi.org/10.48550/arXiv.1506.01497

Ronneberger, O., Fischer, P., & Brox, T. (2015). *U-Net: convolutional networks for biomedical image segmentation*. arXiv. https://doi.org/10.48550/arXiv.1505.04597

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., & Fei-Fei, L. (2015). ImageNet large scale visual recognition challenge. *International Journal of Computer Vision, 115*(3), 211-252. https://doi.org/10.1007/s11263-015-0816-y

Simonyan, K. & Zisserman, A. (2015). Very deep convolutional neural networks for large-scale image recognition, In *Proceedings of the 3rd International Conference on Learning Representations*. https://doi.org/10.48550/arXiv.1409.1556

Song, J., Gao, S., Zhu, Y., & Ma, C. (2019). A survey of remote sensing image classification based on CNNs. *Big Earth Data, 3*(3), 232-254. https://doi.org/10.1080/20964471.2019.1657720

Stetco, A., Dinmohammadi, F., Zhao, X., Robu, V., Flynn, D., Barnes, M., Keane, J., & Nenadic, G. (2019). Machine learning methods for wind turbine condition monitoring: A review. *Renewable Energy, 133*, 620-635. https://doi.org/10.1016/j.renene.2018.10.047

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., & Anguelov, D. (2015). Going deeper with convolutions, In *Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition* (pp. 1-9). IEEE. https://doi.org/10.1109/CVPR.2015.7298594

Wagenaar, B. H., Augusto, O., Ásbjörnsdóttir, K., Akullian, A., Manaca, N., Chale, F., Muanido, A., Covele, A., Michel, C., Gimbel, S., Radford, T., Girardot, B., Sherr, K., Manuel, J. L., Hicks, L., Mahumane, A., Pfeiffer, J., Gloyd, S., Cuembelo, F., ... with input from the INCOMAS Study Team. (2018). Developing a representative community health survey sampling frame using open-source remote satellite imagery in Mozambique. *International Journal of Health Geographics, 17*(1). Article 37. https://doi.org/10.1186/s12942-018-0158-4

Wagner, F. H., Dalagnol, R., Tarabalka, Y, Segantine, T. Y. F., Thome, R., & Hirye, M. C. M. (2020). U-Net-Id: An instance segmentation model for building extraction from satellite images—Case study in Joanopolis City, Brazil. *Remote Sensing*, 12(10), Article 1544. https://doi.org/10.3390/rs12101544

Wigton, W. H. & Borman, P. (1978). A guide to area sampling frame construction utilizing satellite imagery. Technical Report to the United Nations, March 1978. https://www.nass.usda.gov/Education_and_Outreach/Reports,_Presentations_and_Conferences/GIS_Reports/AGuidetoAreaSamplingFrameConstructionUtilizing.pdf

Zeiler, M. D. & Fergus, R. (2014). Visualizing and understanding convolutional neural networks. In D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars (Eds.). *Computer Vision - ECCV 2014. Lecture Notes in Computer Science, 8689* (pp. 818-833). Springer. https://doi.org/10.1007/978-3-319-10590-1_53

Zhu, X. X., Tuia, D., Mou, L., Xia, G.-S., Zhang, L., Xu, F., & Fraundorfer, F. (2017). Deep learning in remote sensing: A comprehensive review and list of resources. *IEEE Geoscience and Remote Sensing Magazine, 5*(4), 8-36. https://doi.org/10.1109/MGRS.2017.2762307

Zhang, L., Zhang, L., & Du, B. (2016). Deep learning for remote sensing data: A technical tutorial for the state of the art. *IEEE Geoscience and Remote Sensing Magazine, 4*(2), 22-40. https://doi.org/10.1109/MGRS.2016.2540798

# Appendix A

## Technical Description of Convolutional Neural Networks

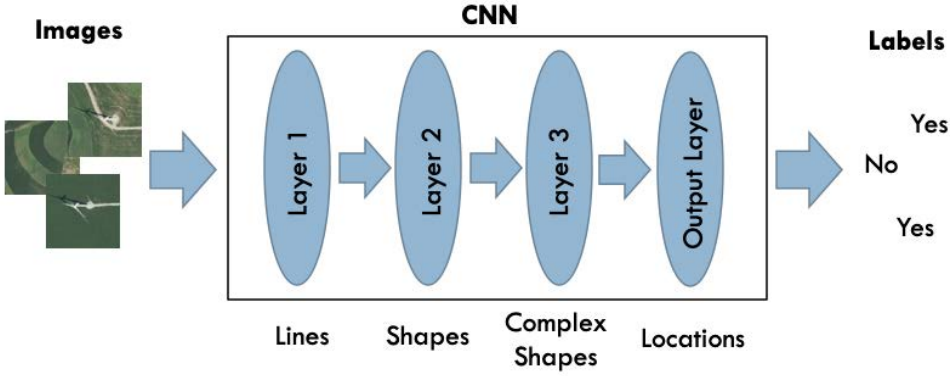



*Figure A1* Convolutional Neural Network Illustration

Convolutional neural networks (CNNs) are a special type of neural network, which are machine learning models that are inspired by the organization of the human brain. Neural networks consist of multiple neurons, each of which is represented by a non-linear regressor that takes as input a vector of numbers and outputs a single value. Neurons are organized into layers, where the outputs of the neurons in one layer are fed as inputs into the next layer. The neural network begins with a layer that takes inputs from the original real-world data and ends with a special output layer that produces the desired predictions. In a

CNN, illustrated generally in Figure A1 (reproduced from Figure 1), the layers learn a hierarchy of information from the pixel data contained in images (i.e., the colors of each dot). The first layer frequently learns how pixels form lines or contain relevant colors, the second layer learns how lines organize into shapes or patterns of colors emerge, the third layer might learn how simple shapes and color patterns form complex shapes, and so on for all remaining layers until ultimately the final layer learns how different types of content organize based on the higher-level abstractions learned in the network.

The parameters to a CNN are the weights of each neuron (i.e., local regressor). Training (equivalently learning) a CNN is the process of fitting all the neurons' weights to the example images, very similar in spirit to fitting the weights of a linear or logistic regression model. Fitting all weights in the CNN involves repeatedly passing each example image (in total called the training set) through the model and calculating the predicted label that would currently be output for each image. Then, stochastic gradient decent is used to update the weights throughout the network to reduce the error in its predictions of all example images. Each pass through the training set is called one epoch, and training continues until either (a) the weights have reached a local optimum from which they cannot be improved, or (b) a fixed number of epochs have occurred so that training does not continue forever (in case the weights never converge but steadily, slowly oscillate).

When employing CNN models for recognizing objects in images (especially in image classification tasks), researchers often have two options. First, for many CNN models, it is possible to download a pre-trained model that already knows how to identify objects from a different data set (e.g., ImageNet composed of 1,000 common objects (Deng et al., 2009)). These models can then be retrained to learn how to identify the researcher's objects of interest. Alternatively, it is also common to train a new model from scratch, reusing the model architecture (e.g., the number of neurons and layers) from a previous study but learning only on data relevant to the researcher's task. The former option has the advantage that it requires less overall training time (fewer computational resources) and leverages information learned that generalizes across many objects of interest (e.g., how pixels organize into lines, how lines organize into shapes, etc.). On the other hand, the second option has the advantage that it does not need to "unlearn" information irrelevant to the researcher's objects of interest, and training from scratch follows the familiar practice of social science researchers when using machine learning models outside of image recognition tasks.

For further information about CNNs and how they might be applied to social science data, please refer to Goodfellow et al. (2016) and De Veaux and Eck (2021).

# Appendix B

## Hyperparameter Tuning Pre-Experiment

In this appendix, we describe the process we used to find the optimal hyperparameters used to train the CNNs for our case study, listed in Appendix C. This was an important step in preparing our experiments because different hyperparameters control different aspects of the learning process. Suboptimal hyperparameter values are likely to lead to weakly performing models. In order to better understand the specific impacts of these hyperparameters and tune our settings to learn the best possible models, we conducted a pre-experiment described here.

For the ZFNet and VGG-16 image classification models, the two hyperparameter we tuned were (1) the learning rate used to update the weights of the neural networks during each epoch, and (2) the under-sampling ratio measuring the number of "no" example images included for each "yes" example image. Setting an improper learning rate could cause the models to learn a badly fitting set of weights in the CNN, so that the model makes more errors when predicting labels than it would with a better set of weights. Using an improper under-sampling ratio could cause the models to become biased towards predicting either label too frequently, causing either over- or under-coverage in the resulting sampling frames.

For the U-Net image segmentation models, we also tuned two hyperparameters: (1) the learning rate, just as with ZFNet and VGG-16, as well as (2) the number of pairs of layers to include in the neural network. Setting an improper number of layers could cause the models to either overfit with too many layers and simply memorize the appearances of windmills in some images, rather than learning general patterns, or fail to learn enough abstractions from the example data with too few layers so that patterns explaining the appearances of windmill and non-windmill images are not found.

In order to find the appropriate values of these hyperparameters, we performed grid searches where different combinations of hyperparameters were evaluated, and the best combination chosen for each of the three types of CNNs considered in our study. For the learning rate hyperparameter, we considered 19 possible values: between 0.00001-0.00009 in 0.00001 increments, between 0.0001-0.0009 in 0.0001 increments, and 0.001. For the under-sampling ratios to use, we considered three possible values: 1, 5, or 9 "no" example images for every "yes" example. As a result, our grid search involved 19 * 3 = 57 hyperparameter combinations for the image classification models learned using ZFNet and VGG-16.

For the number of pairs of layers hyperparameter for U-Net, we considered values of 2, 3, 4, and 5. This resulted in 19 * 4 = 76 hyperparameter combinations for the image segmentation models learned using U-Net.

We evaluated the performance of each hyperparameter combination for all three approaches as follows. First, we took all windmill images for the entire state of Iowa, and we matched them with randomly sampled non-windmill images according to the desired under-sampling ratio (a fixed value of 1 for U-Net, as described in the subsection titled "Step 3: Train a CNN Model"). We then randomly split these images so that 60% were used for training a model, 20% for validation during training, and the remaining 20% were used for evaluating the quality of the model. This was repeated 30 times so that multiple models with different training/validation/evaluation data using the same hyperparameter combinations were created. This reduced the variance caused by the random data splits and created a more accurate estimate of the general performance expected with each hyperparameter combination. The resulting models were evaluated based on their balanced accuracy on the evaluation sets.

We present the average balanced accuracy across the 30 different models for each hyperparameter combination for the ZFNet, VGG-16, and U-Net approaches in Figures B1-B3, respectively. For each figure, we provide both (a) the full charts where the y-axis ranges from 0.0-1.0 (i.e., the full range of possible values of balanced accuracy), as well as (b) a zoomed in version of the same charts where the y-axis is changed to range from only 0.9-1.0 so that the differences between hyperparameter combinations are better visualized. In all figures, the learning rates considered are listed along the x-axis, whereas the under-sampling ratio or number of pairs of layers is given as separate bar charts.

The optimal hyperparameter combinations are represented by the tallest single bar in each of Figures B1-B3. From Figure B1, we find that the best hyperparameter combination for ZFNet is a learning rate of 0.0002 and an under-sampling ratio of 9. From Figure B2, we find that the best hyperparameter combination for VGG-16 is also a learning rate of 0.0002 and an under-sampling ratio of 9. Finally, from Figure B3, we find that the best hyperparameter combination for U-Net is a learning rate of 0.00007 and using 4 pairs of layers. These then became the settings we used for our evaluation experiment (cf., subsection "Step 3: Train a CNN Model" and Appendix C).

More generally, these figures also demonstrate how sensitive the different CNN approaches are to their hyperparameter settings. Both ZFNet and VGG-16 had relatively stable trends in their performance as the learning rate was varied – balanced accuracy slightly decreased as the learning rate reduced from its optimal setting (for each under-sampling ratio), but sharply dropped off once the learning rate became too large (starting around 0.0007-0.0008). Thus, the models are less sensitive to smaller learning rates, but too large of a learning rate could cause poor learning.

For the under-sampling ratio hyperparameter, we see that both ZFNet and VGG-16 benefited from ratios greater than 1, whereas 5 and 9 produced relatively similar performances with slight variations per learning rate (until the learning

rate became too high, then learning was generally unstable). In practice, a value of either 5 or 9 would have probably produced similar learning, so the key here is that it was beneficial to include some greater number of "no" examples images without windmills than the number of "yes" example images with windmills. Of course, this ratio shouldn't be increased too high, or else the models would learn to predict "no" every time.

Finally, for U-Net results, the performances followed less stable trends. Increasing the number of pairs of layers generally helped performance, except 5 pairs of layers was often too many, likely resulting in overfitting and hence models that could not generalize to images not seen during training. The learning rate had a larger impact on U-Net performance than the image classification models learned by ZFNet and VGG-16 (possibly due to the complexities of trying to learn more context in the image segmentation models), so hyperparameter tuning for this hyperparameter could be the most beneficial for studies like ours.

Together, we hope these observations provide some guidance for how one might set the learning rate for other applications, especially if it is infeasible to conduct a hyperparameter grid search before learning the CNN models.

Of final note: the balanced accuracies reported here are higher than the balanced accuracies reported in Table 2 in Section 5. In this pre-experiment, we did not evaluate the models on an entire region of interest, but instead retained the same class imbalance in the training and evaluation sets. This caused fewer false positives, and hence a higher overall balanced accuracy. The sensitivity rates (not reported) were also higher in this pre-experiment.

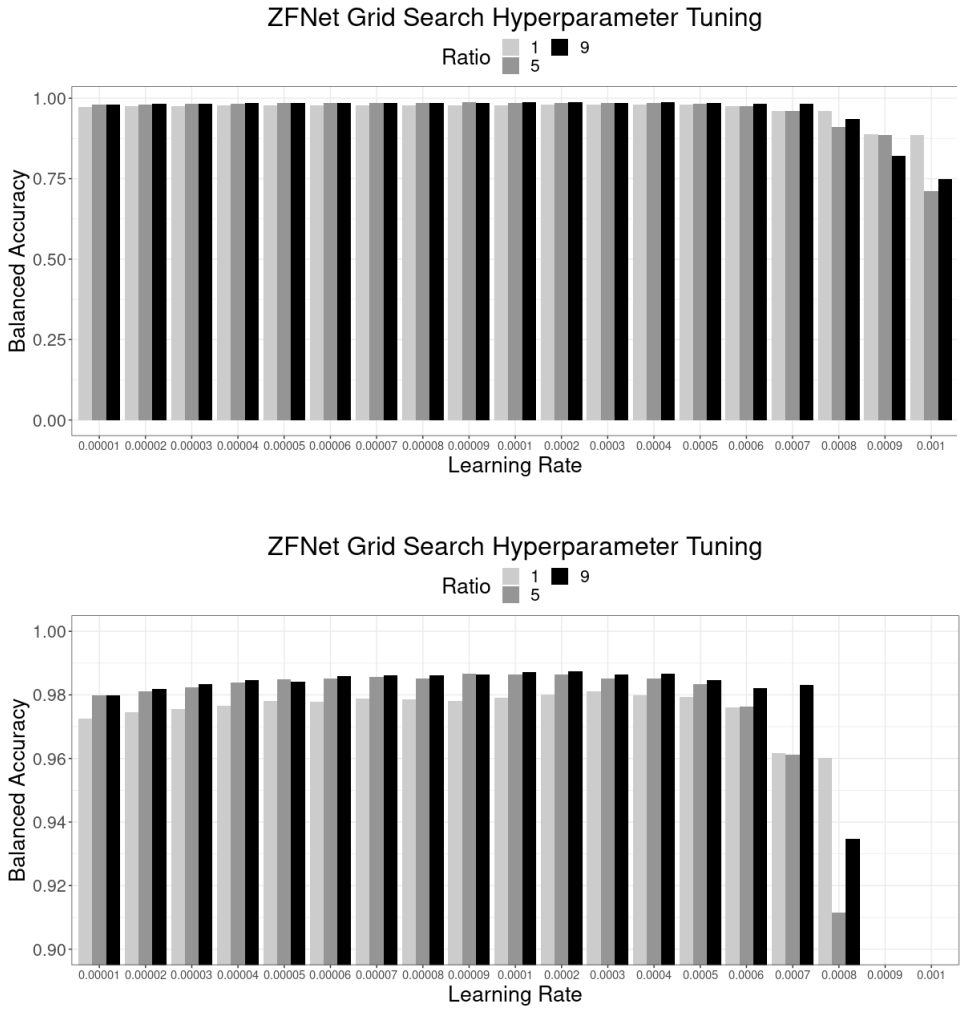



*Figure B1* Balanced Accuracy of ZFNet Models Trained with Different Hyperparameters
(a) Full Charts and (b) Zoomed in on Best Hyperparameter Performances

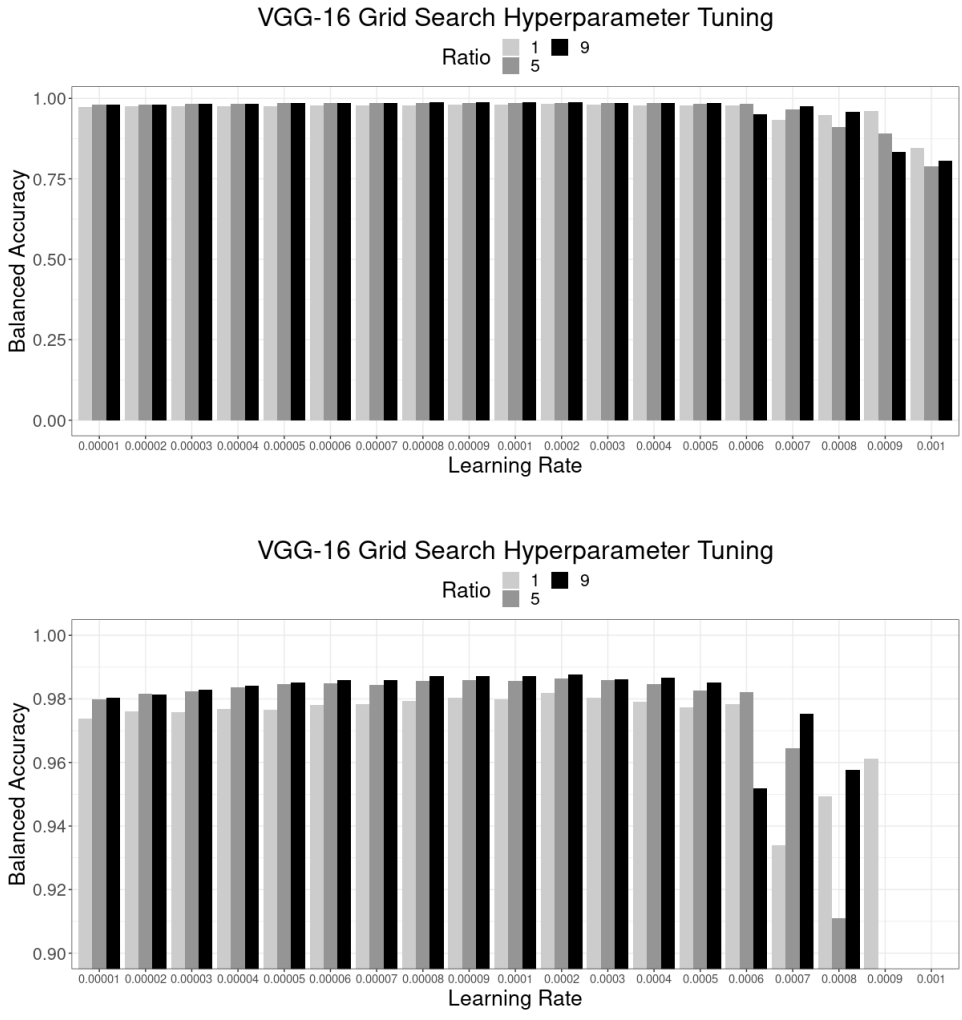



*Figure B2* Balanced Accuracy of VGG-16 Models Trained with Different Hyperparameters
(a) Full Charts and (b) Zoomed in on Best Hyperparameter Performances

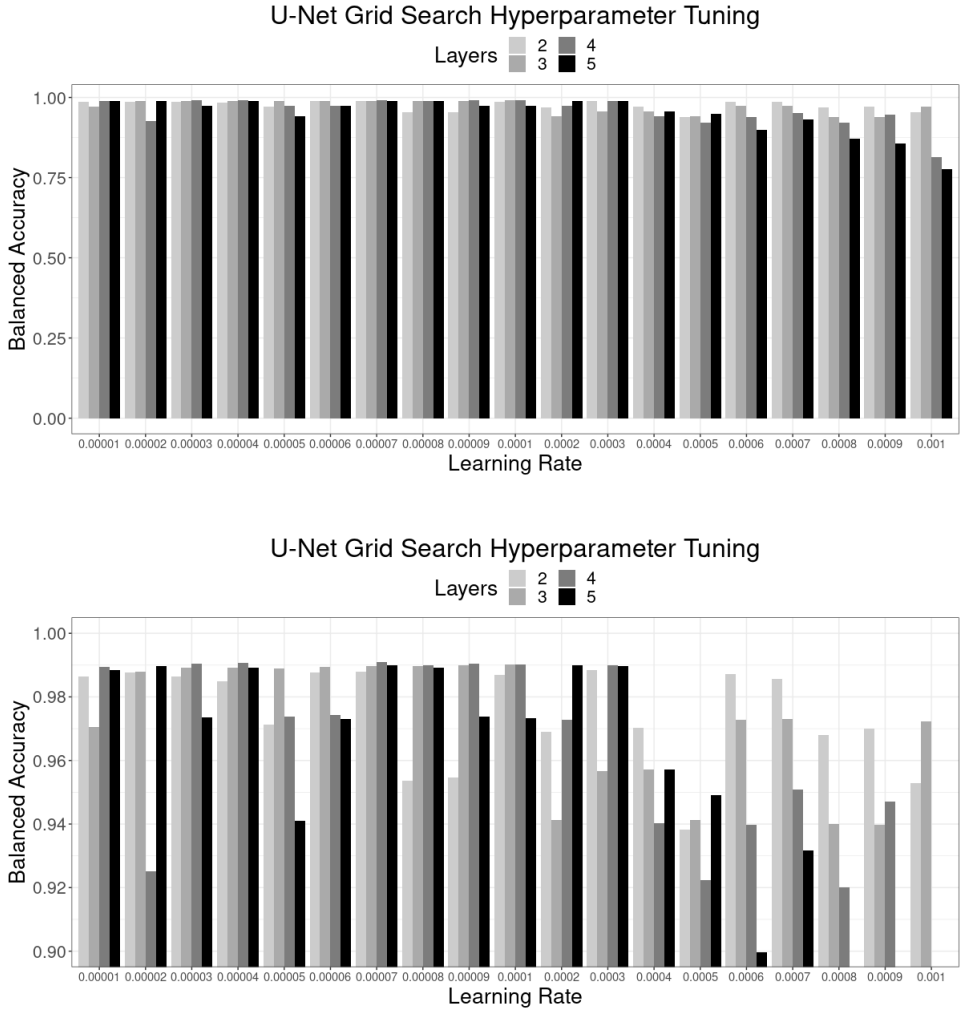



*Figure B3* Balanced Accuracy of U-Net Models Trained with Different Hyperparameters
(a) Full Charts and (b) Zoomed in on Best Hyperparameter Performances

# Appendix C

## CNN Training Hyperparameters

For reproducibility of our evaluation, this appendix provides a description of the key hyperparameters used to learn our CNN models. Here, hyperparameters refer to settings of the learning process that change how learning is performed. We used the state-of-the-art Adam algorithm (Kingma & Ba, 2015) as our stochastic gradient decent process for fitting the weights of the neurons in the CNNs. For the loss function measuring the amount of error in the model's label predictions, we used mean squared error (MSE). 80% of the training set was used for learning. The remaining 20% of the training set was used as a validation set so that the model's performance could be evaluated during training to avoid overfitting. Overfitting is a detrimental side-effect of learning when the model starts memorizing information in the training examples instead of generalizing useful patterns. Training occurred for at most 1,000 epochs, stopping early when validation loss began to increase, indicating overfitting (with a patience of 25 epochs).

The learning rates used (i.e., how aggressively the neurons' weights should be changed during fitting) were tuned for each type of approach using a grid search over possible values. Ultimately, we selected learning rates of 0.0002 for ZFNet and VGG-16 and 0.00007 for U-Net. Finally, we also performed a grid search for the appropriate number of layers to use in U-Net (representing how many levels of visual abstractions the models learn to help identify windmills), settling on 4 as the ideal value. We provide more details about how these values were chosen in Appendix B.

In order to handle the class imbalance issue, we used the popular under-sampling approach to choose the negative examples used from our training set: we performed a simple random sample of negative example images so that the ratio of negative to positive cases matched a desired value. For the ZFNet and VGG-16 image classification models, we chose under-sampling ratios[15] of 9 times as many negative examples as positive examples (details in Appendix B). For the U-Net image segmentation models that learn additional context, any under-sampling ratio of more than 1 caused the models to rarely, if ever, predict the presence of a windmill, so we used an under-sampling ratio of 1 for this approach.

Since under-sampling uses simple random samples of "no" windmill images to build the actual set of examples used to learn a model, this introduces a possible source of variance in the performance of the resulting models. For example, an under-sample that chooses negative examples lacking the full diversity

[15] Given the different under-sampling ratios for the different type of models, we used different probability thresholds from the logistic regressor in the output layers to determine when a label of "yes" should be predicted for an input image: 10% for ZFNet and VGG-16, and 50% for U-Net, each equal to 1 / (ratio + 1).

of non-windmill locations would cause the model to learn fewer useful patterns than a different under-sample that is more diverse. As such, we repeated the approach in our case study 30 times using different random seeds, where different seeds sampled different "no" examples. By averaging the results across these 30 seeds, we can better understand how well the machine learning models learn to identify windmill locations, and as a result, how well our approach performs the task of automating sampling frame construction.

Online First publication date: 30-05-2026