A COMPUTATIONAL APPROACH TO CULTURAL RESOURCE MANAGEMENT: AUTODETECTING ARCHAEOLOGICAL FEATURES IN SATELLITE IMAGERY WITH CONVOLUTIONAL NEURAL NETWORKS

Mallory Willett

TC 660H
Plan II Honors Program
The University of Texas at Austin

May 15, 2019

Alex Walthall, Ph.D.
Department of Classics
Supervising Professor

Adam Rabinowitz, Ph.D
Department of Classics
Second Reader

ABSTRACT

Author: Mallory Willett

Title: A Computational Approach to Cultural Resource Management: Autodetecting Archaeological Features in Satellite Imagery with Convolutional Neural Networks

Supervising Professor: Alex Walthall, Ph.D.

My thesis proposes the use of convolutional neural networks for automatic detection of archaeological features in satellite imagery. Cultural heritage sites require constant management, and archaeologists are increasingly turning to satellite imagery to identify and monitor sites from afar. Given the huge amount of visual information present in these images and the amount of time it takes to do this job with the human eye, I propose a different approach for identifying and mapping archaeological features: using computer vision, specifically an algorithm called a convolutional neural network, or CNN. By training a CNN on a labeled set of hundreds of the same class of archaeological features in a landscape, the CNN can learn to identify new instances of the same class of features in previously unseen satellite imagery. This approach reduces the amount of labor required by analog approaches to feature extraction or traditional survey, and allows archaeologists to more swiftly identify and therefore protect areas of cultural significance. My research on CNNs in other fields and inroads made on a proof-of-concept CNN to identify archaeological features demonstrate the feasibility of using this type of algorithm to automatically detect archaeological features in satellite imagery.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS

To my my supervisor, Dr. Alex Walthall, and second reader, Dr. Adam Rabinowitz, thank you for your guidance, assistance, and motivation over the course of this project, for pushing me to pursue a topic outside of my comfort zone and for encouraging me to do my best work. Thank you to Dr. Qixing Huang and Arjun Karpur in the Computer Science department at the University of Texas at Austin and to Dr. Mujeeb Basit at UT Southwestern Medical Center for their guidance and assistance on the coding portion of my research, and to Bogdan Șandric and Carol Opriș at the National Institute of Heritage of Romania (CIMEC Department) for their work on gathering information about the tumuli in Dobrogea and for graciously allowing me to use their data. Thank you to my friends who listened to me talk about this project incessantly over the past nine months, and thank you to my parents, who provided me with a perfect balance of emotional support and tough love during the times when completing this thesis felt like an impossibility. This project could not have happened without you all.

INTRODUCTION

My relationship with archaeology began by accident. During my sophomore year, I built a code for a project in machine learning club using the neural networks API KERAS, named after the Greek word for horn, κέρας, a word first found in Homer's *Odyssey* (https://keras.io/#why-this-name-keras). Elated by the connection between the two fields I found most interesting—Classics and Computer Science—I brought the coincidence to my then Greek professor. It was my second semester pursuing my Classics major, and I had not yet realized that there were people who maintained a professional interest in both Classics and Computer Science. A year later, I would read a book of the *Odyssey* in its original Greek and, finding the word *κέρας*, smile at how I had grown as a student with greater proficiency in both fields. This time, I showed the name of this API to my professor, and he, learning that I had coding experience, responded by asking if I would like to work on the database for his excavation that summer. My relationship with archaeology may have begun by accident, but it has continued on purpose. The following summer, I excavated on Crete with the Azoria project, and the next, in Athens with the American School's Agora Excavations. During each intervening school year, I continued to pursue my studies in both Classics and Computer Science, thinking about the ways that I could apply the skills I was developing in the realm of computer science to the archaeology that I was doing in the summers. So, I thought and I thought some more about how I might connect these two fields that I loved, and gradually my nebulous ideas for how I could synthesize my two interests found their footing, leading to the thesis you are reading now.

My primary driving force for this project was to marry computer science and archaeology and, ultimately, to create a thesis that was not reflective or summative but forward-looking and immediately useful. There is plenty of work in digital humanities, and digital classics in particular, musing on the state of the quickly-digitizing field calling for serious reflection on what we want archaeology to look like in the decades to come (Bill Caraher's work on "slow archaeology" comes to mind). And this is all valuable and important work – the nature of archaeology can change when we introduce digital tools into the field. A less-than-careful approach to digitizing the practice of archaeology, including how we record our data, how we think about the things we dig up and how we experience the very act of excavating, can seep into the questions we ask as archaeologists and the conclusions we draw from our data (Caraher 2016:423).

But, as an undergraduate student, and one who has a vested interest in computer and data science, I felt the best way to make a meaningful contribution to the field while also utilizing and improving my skills was to propose an application of computer science for archaeology and to document my process of doing so. What you will read in the following 50-something pages, then, is my proposal for the use of convolutional neural networks in the fields of cultural resource management and archaeology. Specifically, what I will propose is the feasibility of automizing large-scale feature extraction from satellite imagery by leveraging convolutional neural networks.

The task of large-scale feature extraction is a time consuming one, and one that has been approached in many ways, both digital and physical, within and without the field of archaeology (see, for example, Abolt, *et al.* 2019; Albert, *et al.* 2017; Lin, *et al.* 2014; Schuetter, *et al.* 2013). A manual approach to identify unknown archaeological features in satellite imagery across Mongolia, for example, took some 30,000 human hours (Lin, *et al.* 2014). Physical large-scale survey efforts also require manpower and the associated investment of time and resources, which vary based on a number of factors including the level of training and effectiveness of field personnel (Banning, *et al.* 2017). This time commitment, coupled with threats of urban sprawl, climate change, and armed conflict, contribute to the urgency of finding a faster way to identify archaeological sites and features on a large scale so that those sites might be better protected from these threats. The work presented herein, then, holds value not as a timesaving measure or as a way to save costs by keeping boots off the ground, but for its promise of identifying sites of cultural value wherever we have remote sensing data, especially those sites at immediate risk of destruction.

This thesis includes background research and an assessment of the state of the fields of archaeology and remote sensing and of machine learning and convolutional neural networks. It also includes my own proof-of-concept code which aims to demonstrate the feasibility of my proposed approach, and a reflection on the choices and mistakes I made, which I hope will both advance the body of knowledge on this subject and make life a little bit easier for the next person who approaches a task like this. My proof-of-concept code attempts to automatically detect tumuli, or burial mounds, in satellite imagery of the ancient Greek city of Histria on the Black Sea coast of Romania. I hope that the information contained herein is valuable to classicists and computer scientists alike, and that it will be used thoughtfully in the promotion of digital approaches to archaeology.

A NOTE REGARDING TERMINOLOGY

Throughout this paper, I employ several important terms and phrases that may not be familiar to the reader. Thus, it is essential that I attempt to define these terms and phrases from the outset, so that when the reader will be well-informed of *my* definition of these terms when they are first encountered in the text. The first phrase is "cultural resource management," often abbreviated as CRM. I will use the phrases "cultural resources," "cultural heritage sites" and "cultural heritage monuments" interchangeably. Cultural resources and cultural heritage sites refer, in this paper, to ancient sites of cultural and historical significance. Their management, then, includes the process of excavating and ensuring their protection by navigating the changing political, urban and ecological climates that may threaten their existence (White and King 2007:141-142). Archaeology, then, is a form of cultural resource management, insofar as excavation involves interacting with and negotiating for the protection of these sites. What happens to heritage sites after excavation, especially those under threat, as this paper will discuss, also falls under the umbrella of cultural resource or heritage management.

The next phrase to clarify is "computer vision." Here, I am not referring to when your paranoid friend claims that the NSA guy is watching you through your laptop camera. Rather, computer vision involves a whole host of tasks that the computer can do that would typically require human sight. These tasks include identifying objects in an image and tracking objects as they move in a video; a number of computer vision approaches have been developed, for example, as a way to track cars in video surveillance in real-time, as a response to increasing traffic (Coifman, *et al.* 1998; Anandhalli and Baligar, 2018). Computer vision played a role in the missions of Mars rovers Opportunity and Spirit, helping the rovers detect obstacles and increase the distance traversed each day by the rovers (Matthies, *et al.* 2007:74-76). Computer vision is also behind Facebook's DeepFace algorithm, powering the company's scarily accurate ability to suggest who it is in that photo you just uploaded (Taigman, *et al.* 2014). Essentially, any time a machine interprets an image, we are dealing with computer vision.

Readers may be familiar with traditional neural networks, which interpret text data. A *convolutional* neural network, or CNN, is a type of neural network that performs computer vision tasks, and is the kind that I have employed in my thesis project. Built with image processing tasks in mind, CNNs – in contrast to traditional neural networks – take images as input, where each pixel (rather than, say, each word) is a relevant feature of the data (Burkov 2019:66). Convolutional neural networks, or CNNs, are called "neural" because they were modeled after the way the human brain learns to interpret new visual information, where blocks of pixels are neurons and they talk to each other in a web of mathematical functions that spit out a classification at the end.

CNNs are algorithms that perform "machine learning," which is a phrase applied by computer scientists to situations where algorithms make predictions from data and change to improve their performance on a given task (Burkov 2019:1)[1]. Readers might be familiar with AlphaGo, the first computer program to beat a human at the board game Go– a game considered much harder for a computer to win than chess – without any handicaps. AlphaGo's ability to start as a *tabula rasa,* with no experience or domain knowledge of the game of Go, and achieve dominance greater than the best human player of the game in only 24 hours of playing against

[1] Andriy Burkov's book *The Hundred-Page Machine Learning Book*, which I reference several times in this paper, is an excellent resource for those looking to learn more about machine learning algorithms. It is technical without being opaque, accessible for those with little previous exposure to the subject, and concise.

itself is a classic example of machine learning (Silver, *et al.* 2017:1-2). How the applications of machine learning and CNNs can be beneficial to archaeological investigations will be dealt with at greater length later in this paper.

Cultural resource management is a global enterprise (King 2011). The conservation efforts required by sites around the world differ wildly – papyri recovered from the dry sands of Egypt pose entirely different challenges than preserved bodies uncovered in the peat bogs of England and Denmark, for example. But the basic factors that threaten destruction for cultural heritage sites are largely ubiquitous – and, it is worth noting, usually anthropogenic.

Urbanization ranks prominently among these threats. As cities grow, they encroach further on these sites. Infrastructure initiatives and population growth bring cars, people and industries closer to archaeological and heritage sites, putting significant pressure on these irreplaceable spaces. For example, a team of scholars from the University of Cyprus studied the effects of urbanization on cultural heritage monuments in the Paphos district of Cyprus, choosing this location as a place that is both home to many cultural heritage sites, some of which are UNESCO World Heritage sites, and which has experienced massive urbanization in the past 35 years (Agapiou *et al*. 2015). Based on their research, the team found that urbanization in an area like Paphos poses threats for the excavated, visible monuments: construction and heavy traffic create vibrations that are destructive to the structural integrity of nearby monuments; pollution from vehicles coming nearer to the site damages the monuments which, in Paphos, are constructed from highly porous rock that absorbs particles in the air. Moreover, they found that urbanization also poses threats to still unexcavated archaeological materials, as rapid construction to meet the needs of a growing population can destroy these hidden features. We hear about this in cities like Rome, where subway construction often reveals the foundations of some ancient wall that is then preserved, but smaller and less metropolitan places like Paphos are also home to a wealth of cultural heritage materials that are at risk for damage or destruction from urbanization.

Climate change poses another threat. Small upward trends in temperature may not present an immediate or drastic threat to cultural heritage materials; however, this temperature increase changes the incidence of freeze-thaw events, which are important to the preservation of cultural heritage monuments. A team studying how climate change will affect freeze-thaw events in Europe found that even slight changes in temperature can alter the number of freeze-thaw cycles in a given period, which in turn affects the rate of deterioration for cultural heritage sites exposed to these freeze-thaw cycles. For sites in the far north preserved in permafrost, an increase in the number of freeze-thaw cycles can upset the preservation of the soil and deteriorate stone, increasing the risk of damage to the archaeological and paleoecological remains preserved therein (Grossi *et al*. 2007). Climate change threatens archaeology in other parts of the world, too, notably coastal areas: a study found that if the current global temperature is sustained for the next two millennia, 40 UNESCO world heritage sites will be affected by the corresponding rise in sea levels, since sea levels continue to rise in response to sustained high temperatures; if temperatures increase by 3 degrees Celsius, that number goes up to 136 sites (Marzeion and Levermann 2014:4). And it bears reiterating that these are just the ones listed as world heritage sites by UNESCO – countless more cultural heritage sites located along coastal areas face the same fate as temperatures and sea levels continue to rise (Marzeion and Levermann 2014:7). But climate change along coasts will not affect all areas equally, due to variances including the composition of the ground and the interaction with the tides. Climate change will affect inland areas, too, as people living in affected coastal areas move inland, increasing development inwards from the shoreline and adding pressure on inland heritage sites and extending the zone at risk of destruction (Anderson *et al.* 2017:11). Those working in cultural resource management must deal urgently with the threats posed to archaeological sites by the changing climate.

Armed military conflict looms as perhaps the most dramatic threat to the preservation of cultural heritage sites. One has only to look to the news of the past few years for evidence. In fact, the list of cultural monuments destroyed by ISIL alone is large enough to warrant its own Wikipedia page (and not just a stub!: https://en.wikipedia.org/wiki/Destruction_of_cultural_heritage_by_ISIL). In addition to destruction, conflict brings an increase in looting, too (although looting happens frequently even in areas without active conflict). It is difficult to quantify the amount of conflict-related looting and destruction happening at archaeological sites around the world. In part, this is due to the fact that, as Blythe Proulx has noted, "there is no 'master catalogue' of all archaeologically significant sites around the world both known and unknown, so the task of assessing the extent of the damage caused by looting around the world is difficult" (Proulx 2013:111). Furthermore, most work on the relationship between conflict and looting in those areas most affected is anecdotal and qualitative, taking the form of journalistic pieces or case studies tracing looted objects to the market. Nevertheless, there is work that suggests a long-term positive correlation between looting and armed conflict (Fabiani 2018:3-6)

Conflict poses an additional burden to cultural resource management because it obstructs archaeologists and conservationists from having physical access to a site. In areas of active conflict or looting, it can be dangerous for archeologists to monitor, much less intervene in or protect, sites under threat. In fact, initial in-person assessments of archaeological sites even in areas without armed conflict can be difficult and infrequent due to the degree of human resources required at large sites (Cleere *et al*. 2016:3). Monitoring cultural heritage sites for changes requires both a baseline knowledge and mapping of a given site and the ability to ground-truth the current state of the site against its former condition, meaning archaeologists must gather large amounts of unbiased data regarding the physical extent of the site's structures and their rate of deterioration before any fruitful ongoing monitoring can occur (Cleere *et al.* 2016:7). Doing this job in person is expensive and slow, requiring manpower and all the costs that come with it. The combined weight of all of these factors, coupled with the advantages of remote sensing for monitoring sites on a landscape level, may explain why archaeologists are increasingly turning to satellite imagery and remote sensing in their effort to monitor and combat the loss of cultural heritage sites around the globe.

Archaeology and remote sensing have shared a close relationship for a century. Remote sensing owes its birth as a scientific field to aerial photography prior to and during World War I, when military pilots from various countries took photos of the ground from above. Though primarily for military purposes, the value of aerial photography for archaeology was immediately noted – one such pilot, UK army Lieutenant P.H. Sharpe, earned the moniker "pioneer of aerial archaeology" for being the first to photograph Stonehenge and its surrounding plain from above when wind blew him off his military course (Capper 1907:By the end of the 1930s, archaeologists recognized the advantage of height in aerial photography and were organizing flights over Europe, the Middle East, India, and Central America to collect photography for the purposes of finding new archaeological discoveries (Parcak 2009:14-15).

By the Second World War, major world militaries were taking aerial photographs of most of Europe to aid in military reconnaissance, and archaeologists themselves collected aerial photographs in the interest of documenting and protecting historical sites. During the war, many archaeologists learned to work with aerial photos through the field of intelligence. The first use of infrared film in aerial photography came in the 1950s, bringing the archaeological features therein into clearer focus, and with the sixties came satellites and space photography, as the US funneled funds into space technology on the heels of the Soviets' launching of Sputnik (Taubman 2003:212). Satellite imagery from space reconnaissance satellites like the Corona, Argon, and Lanyard systems, originally sent into orbit to gain military intelligence on the Russians, would become declassified 25 years later, and that imagery would bring a windfall of archaeological discoveries (MacDonald 1995:694, 698).

The seventies were marked by a transition to multispectral satellite photography and the launching of the Landsat satellite system by the US Department of the Interior, which spurred a number of studies with applications for archaeology in the following decade. For instance, the First International Conference on Remote Sensing and Cartography in Archaeology was held in 1983. And in the following year, NASA sponsored a congress on remote sensing in archaeology, the impact of which Sarah Parcak writes "cannot be overstated". One NASA archaeologist, Tom Sever, wrote in a report on the meeting:

> New technologies to which [archaeologists] were introduced may represent the kind of scientific breakthrough for archaeology in the second half of the 20th century that radiocarbon dating was in the first half of the century ... advancements in these areas are occurring so fast that unless archaeologists apprise themselves of the technology now, they will be unable to keep pace with the technology in the near future (Sever and Wiseman 1985: 2–11).

Satellite archaeology continued to grow in popularity from the 1990s onwards, especially as major organizations like UNESCO endorsed satellite imagery and remote sensing as tenets of cultural resource management. More recently, several high-resolution satellites have been launched, bringing resolution of these photos down from sometimes 15 meters per pixel to as precise as 0.61 meters per pixel, in the case of the QuickBird satellite, or 0.41 meters in the case of GeoEye-1 (“LAND INFO” 2018). These improvements advanced the utility of satellite archaeology from identifying large-scale features and patterns to identifying even smaller objects in the landscape.

Because satellite imagery advancements have reached such a high ceiling for image resolution, it is possible, with good enough imagery and a good enough algorithm, to train a computer to automatically detect features in that imagery with a high degree of accuracy. A quick Google search for machine learning reveals that, although the topic boasts a massive buzz, it is difficult to concisely define. NVIDIA, a producer of computer chips and graphics processing units, cleanly defines machine learning as “the practice of using algorithms to parse data, learn from it, and then make a determination or prediction about something in the world.” It differs from other forms of computer programming in that the machine is “‘trained’ using large amounts of data and algorithms that give it the ability to learn how to perform the task,” rather than explicitly coded with specific instructions (Copeland 2016). Convolutional neural networks, as I will discuss in a few pages, are one type of algorithm used to perform machine learning tasks, and I will demonstrate how I am training this type of algorithm to extract tumuli from satellite imagery on page 43.

Machine learning tasks broadly fit into one of three categories: supervised, semi-supervised, and unsupervised. The end goal for all of these categories is to predict accurately the outcome variable for a set of data for which that variable is currently unknown. If there is data for which the outcome variable is already known and labeled, we can use these instances to “supervise” the machine learning process. As the algorithm makes predictions on the unlabeled data, it is iteratively corrected by the known, labeled data until its accuracy reaches an acceptable threshold, which a computer scientist can decide based on domain knowledge and the given task (Burkov 2019:5).

Classification is the most commonly task performed through supervised learning, and this is the task that the algorithm in my thesis attempts to perform. A classification algorithm attempts to predict what category some entry in the dataset will fall into, based on the other known variable quantities associated with that entry. Classification can be done on traditional text-based data – will, for example, a customer default on a loan? The bank can classify that consumer as likely or unlikely to default on a loan based on the other variables whose quantities are already known like age, marriage status, or income. Classification can also be done on images; the textbook dataset for this kind of problem is called the MNIST dataset. This dataset contains tens of thousands of images of handwritten digits and is often used as a starting point for people who want to learn classification without spending too much time processing messy or cumbersome data. (Interested readers will find no shortage of "Build A Classifier in 10 Minutes!"-esque how-to articles on the web that are built upon this dataset.) This was the first classification algorithm I ever built, and I used the bones of my MNIST algorithm to build the code for this project.

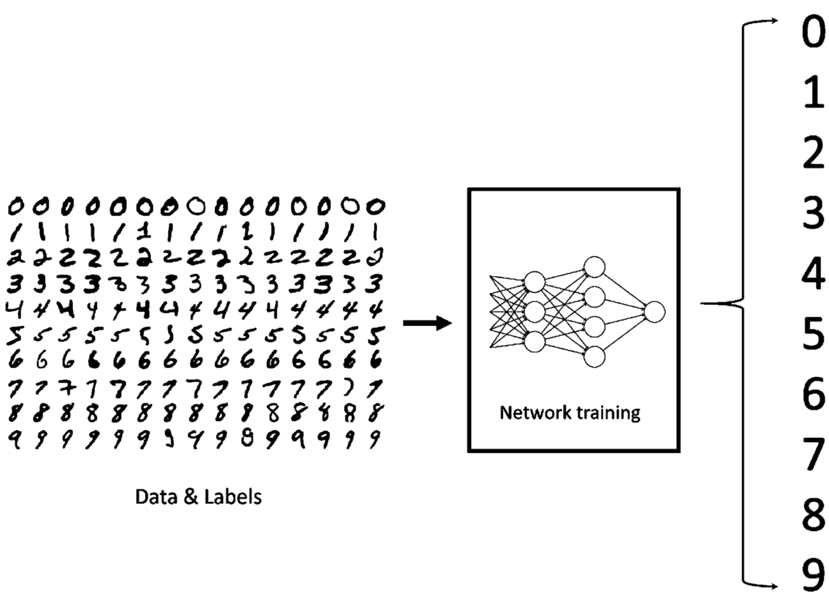

Figure 1: A bare-bones visualization of the MNIST dataset, algorithm and output, from https://towardsdatascience.com/image-classification-in-10-minutes-with-mnist-dataset-54c35b77a38d

Unsupervised machine learning attempts to make sense of unlabeled data, or data without labeled outcome variables. Here, algorithms attempt to discover the underlying structure or grouping of the data without explicitly telling you what that data *is,* as in the case of classification. Clustering is the most typical unsupervised machine learning task, in which algorithms do their best to identify intrinsic groupings in the data.

Semi-supervised machine learning falls somewhere in between the two, where the outcome variables for some of the data are labeled but some are unknown. In this case, a combination of supervised and unsupervised techniques can be used to identify the structure or classification of the data whose outcome variables are unknown (Brownlee 2016). Semi-supervised learning proves useful in many real-world machine learning examples where the amount of data is too massive for each instance to have its own label, but we nevertheless have some basic grasp of what classes the data might fall into based on a labeled subset of that data. It can also provide gains in accuracy and efficiency over fully supervised learning, because it reduces the amount of human bias implicit in data labeling and also decreases the amount of time spent labeling and computer memory spent processing those additional pieces of information. For example, a computer scientist might want to classify websites based on their text content. That person might have a set of labeled websites, teaching the computer that specific types of websites exist and giving it a framework for what those types of websites might be. The scientist can then input a massive number of unlabeled websites, and this new, unlabeled data will allow the computer to more fully define which websites are more and less similar to each other and perhaps even identify some new types of websites not present in the training set (Castle 2018). Additionally, Amazon's Alexa AI team implemented a semi-supervised approach on Alexa's speech recognition algorithm and reported an error reduction of up to 22% over previous, fully supervised models (Johnson 2019).

Understanding the breadth of tasks that machine learning can accomplish and when to apply which type of learning is crucial for deciding which of these types of machine learning, and which algorithms, are appropriate for tackling a given problem. Having examined the three subdivisions of machine learning, the next section of my thesis will introduce convolutional neural networks and explain why this supervised learning algorithm is best suited to solving the problem in my proof-of-concept code.

Before I can present some applications of convolutional neural networks and the methodology of my code, I would like to talk more about CNNs, how they work, and their standard applications. From algorithms that can extract land boundaries for the purpose of determining the potential profit and insurance costs of a tract of farmland (Babawuro 2012) to seeing if a computer can tell the difference between a chihuahua and a muffin (Yao 2017) to suggesting who you should tag in photos you add on Facebook (Taigman 2014), CNNs have helped make object detection a ubiquitous topic in computer vision (Cao and Choe 2018:3), often achieving remarkable accuracy in object detection and classification problems compared to other supervised learning algorithms (Krizhevsky 2017:8).

Figure 2: Attempting to classify objects with similar appearances using computer vision. From “Chihuahua or muffin? My search for the best computer vision API,” Mariya Yao.

Convolutional neural networks are a type of neural network. You might know something of neural networks in their own right, or perhaps you have heard people use the buzzword “deep learning,” which in fact refers to learning performed by neural networks (Burkov 2019:65). As I briefly mentioned in my note on terminology, and as the name of the algorithm suggests, the architecture of neural networks was inspired by the structure of the human brain. As Kevin Gurney writes in his book *An Introduction to Neural Networks,*

> A neural network is an interconnected assembly of simple processing elements, *units* or *nodes*, whose functionality is loosely based on the animal neuron. The processing ability of the network is stored in the interunit connection strengths, or *weights*, obtained by a process of adaptation to, or *learning* from, a set of training patterns (Gurney 1997:1).

In the human brain, neurons communicate with each other through electrical signals transmitted over axons, and these electrical signals pass over gaps called synapses and are received by dendrites. Each neuron is connected to thousands of other neurons, and the brain sets a threshold for when each neuron should "fire," which is determined by the strength of the synaptic connection between the neurons.

This basic architecture of human learning informs the artificial learning performed by neural network. In a neural network, nodes are equivalent to neurons; weights model synapses and allow the network to decide whether or not a node in the next layer should be activated (nodes in a network will only be activated if they meet some pre-determined threshold, just as neurons in the human brain will only fire if their activation function is met). In the figure below – a standard visualization of the architecture of neural networks – each circle represents a node. Each connecting line will have an assigned weight, and one output node will activate based on which output node reaches the threshold for activation (traditionally, the node that activates will produce an output of 1, indicating a positive classification, and those that do not activate will output a zero, indicating that those possible outcomes are not the correct classification based on the information from the input nodes) (Gurney 1997:1-2).

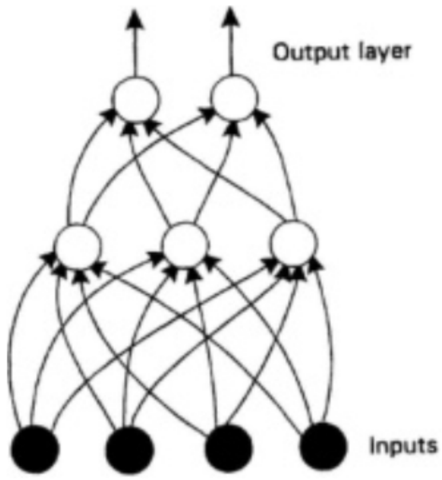

Figure 3: a simple example of a neural network, after Gurney 1997:2.

For example, we can imagine the figure above represents a neural network that would like to classify an email as “spam” or “ham” (non-spam email). Pretend that each node in the input layer, represented by the four black nodes, is a word taken at random from an email. The network will pass those words through a hidden layer, denoted by the three nodes in the center, and the embedded mathematical functions (informed by the weights assigned to the arrows) will produce an output that activates one of two of the output nodes. If the output of the “spam” node is 1 and the “ham” node is 0, the network is telling us that the email is spam. Conversely, if “ham” activates with a value of 1, then the network is classifying this email as ham and not spam. This is the basic way a neural network operates.

If we would like to use the architecture of the neural network above to classify images, the size and cost of the problem would quickly become intractable. This is because, when classifying an image, each pixel is a relevant piece of data for determining the correct output – meaning that each pixel requires its own input node. Even for small images like the ones I use in my own proof-of-concept code, which measure 155 by 155 pixels, this amounts to (155x155=)

24,025 input nodes. Following the same interconnected structure above, the amount of mathematical functions performed by the computer would increase rapidly as layers are added to the network.

While convolutional neural networks follow the same basic structure of nodes and layers depicted in the previous figure, CNNs significantly reduce the number of parameters in models with a large number of input nodes without sacrificing accuracy, making them the appropriate choice for image processing tasks (Burkov 2019:66). The common structure of a CNN includes the input, a convolutional layer, a sub-sampling layer, a fully-connected layer, and an output.

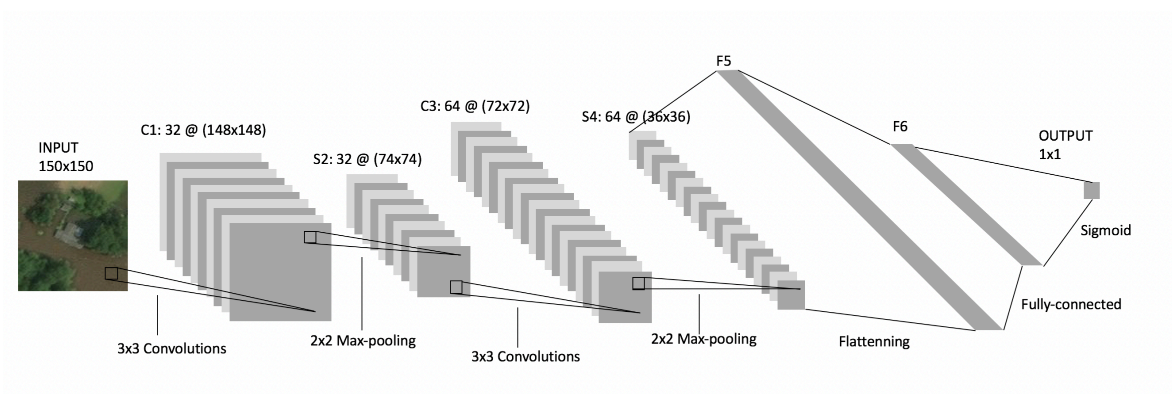

Figure 4: visualization of CNN structure (after Cao and Choe 2018:4).

While all of the hidden (that is, not input or output) layers detailed above involve mathematical processes that combine to produce the final output, it is not strictly necessary to understand this math in order to grasp the basic way a CNN works. Therefore (and because I am not a math major), I will explain only the inner mathematical workings of the convolutional layer, as it is the most significant for understanding the algorithm. I will note, however, that the purpose of all of the hidden layers in a CNN is to identify features throughout the image and reduce the size of input feature matrices, until the final layer flattens those matrices into a single vector which is at last passed through an activation function to produce the output classification (Cao and Choe 2018:4-5).

Convolutional layers reduce the dimensionality of the input image while retaining the salient information contained therein. For example, a 3x3 convolutional layer will pass over the image in a "moving window" approach, focusing on small squares of size 3 pixels by 3 pixels at a time within the original image, which we will call "filters". Each of the (3x3 pixels=) 9 pixels in these filters receives some numeric value based on the visual information contained by that pixel, which is represented in a matrix. The convolutional "moving window," often called a "patch," also takes the form of a 3x3 matrix, with each value therein determined by the computer scientist based on the nature of the feature they would like to extract. The convolution, then, is the mathematical function of performing matrix multiplication on the filter and patch, which produces some sum value. This sum value is passed through an activation function and becomes a neuron value in the next layer of the CNN (Cao and Choe 2018:3-4, Burkov 2019:66-69).

Although CNNs notably yield the highest accuracy in image classification, computer scientists consider other criteria when deciding which machine learning algorithm to apply to a given problem, too, and one of the most important factors is explainability (Yang, *et al.* 2019). Explainability refers to the ability to retrace and understand how an algorithm made the decision that it did. In some algorithms, like decision trees, for example, a human can easily work backwards to understand the decisions that the algorithm made which led to its output; in a another algorithm, the support vector machine, the output is an equation that one needs only read to understand; but in a neural network, while the *output* is easily interpretable – the image does or does not contain a tumulus; the email is or is not spam – it is nearly impossible to explain how the algorithm made its decision, and why it made that decision but not a different one. This is because the algorithm itself is a "black box" (Burkov 2019:48).

Explainability is important in domains where the stakes are very high, like medicine. It is hard to trust and apply the results of an artificially intelligent model when doctors cannot understand how a model came to the conclusion it did and why not some other conclusion, and the possible cost of a mistake is human lives lost. Unfortunately, the two goals of accuracy and explainability are often at odds with one another, where some of the most easily interpretable machine learning algorithms are the least accurate, and vice versa (Yang *et al.* 2019:2). While computer scientists across many domains continue to search for highly accurate artificially intelligent algorithms whose results are also explainable[2], the stakes for misunderstanding how and why my CNN identified the tumuli it did are low, rendering explainability a minor concern for my research. CNNs, then, do not present a significant explainability problem for the purpose of this thesis, allowing me to continue on with this approach due to their greatest potential for accuracy.

[2] Readers interested in the search for "explainable AI" should see DARPA's document on the subject at www.darpa.mil/attachments/XAIProgramUpdate.pdf, particularly pages 2 and 5.

Because satellite imagery has achieved such high resolution, it is possible to identify even small physical features in good imagery. This finds applications in archaeology where archaeological features large and small alike can be identified in high-resolution imagery with either manual or computerized techniques. Satellite imagery proves especially useful for identifying archaeological features because "viewing archaeological structures from ground level generally does not clearly identify the spatial characteristics of these structures of the relationship to surrounding archaeological sites." Furthermore, "in some cases ancient structures are not apparent from ground level but become obvious from birds-eye view" (De Laet *et al.* 2006:830).

With the wealth of satellite imagery provided by platforms like Google Earth and upon request from various satellite services like DigitalGlobe or the United States Geological Service, archaeologists have no shortage of imagery to draw from if they wish to remotely identify or monitor archaeological sites. However, manually extracting archaeological features from satellite imagery can be time consuming, especially as the scale of the job and area covered by the imagery increases. An extreme example of just how time consuming these jobs can be comes from a team at the University of California at San Diego, who endeavored to crowd-source the search for the tomb of Genghis Khan (Lin *et al*.
2014). This search drew on the manpower of ten thousand online volunteers who participated for an average of three hours per person. Although this massive effort did not locate Khan's elusive tomb, it succeeded in identifying dozens of previously-unknown archaeological sites, 55 of which were confirmed on the ground by a National Geographic expedition team. Of course, this effort spanned an enormous ground area, and the task was far more vague than most feature

extraction tasks (participants were not trained archaeologists, and were simply asked to label anything "out of the ordinary"), so the time required was magnitudes larger than most efforts to extract archaeological features from satellite imagery.

It remains the case, nonetheless, that manual efforts to locate, label, and map disparate archaeological features across a vast area are time consuming (and not a terribly stimulating use of that time, either). Even Sarah Parcak, a highly-skilled archaeologist in her own right, still launched the GlobalXplorer project to crowdsource civilian efforts in order to expedite site discoveries in satellite imagery ("About the Global Explorer project" 2018). That automating feature extraction from satellite imagery by employing a convolutional neural network can mitigate such investments of time is well attested in many fields, academic and commercial. A team at the University of Texas at Austin trained a CNN to automatically extract ice wedge polygons from imagery of ice shelves in Alaska (Abolt *et al.* 2019). Project members used the CNN to identify pixels in LiDAR imagery that comprised the edges of ice wedge polygons.

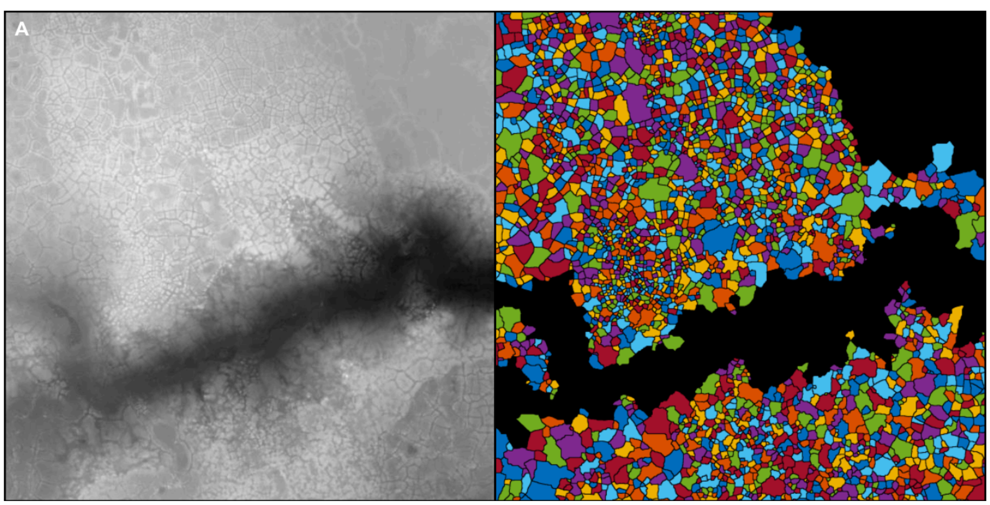

Figure 5: example CNN output from Abolt et al. transforming raw LiDAR data (left) into easily discernable polygons.

This image from their paper stood out to me as an example of just how helpful convolutional neural networks can be for feature extraction and its applications. Counting, mapping, and monitoring changes in ice wedges based only on the LiDAR data in the left-hand image would require an exceptional commitment of time and resources, but the polygons extracted by the CNN on the right are far easier to interpret and manipulate. The results and the broader applications of this project inspired my own work.

Still more applications of CNNs for the social sciences can be found in the work of computer scientist Dr. Stefano Ermon at Stanford. Ermon has published work on socially conscious uses of CNNs and has an academic interest in computational sustainability, including assessing the quality of infrastructure in sub-Saharan Africa using satellite imagery and deep learning techniques, including CNNs (Oshri *et al.* 2018). Monitoring changes in quality of infrastructure on a large scale, much like monitoring changes to archaeological features over a large landscape, is time and resource intensive. Nevertheless, infrastructure quality is a major – and sometimes the only – indicator of quality of life in developing countries where statistical information on the economic status of individuals is poor or nonexistent, making it imperative to be able monitor development in a cost-effective way. Ermon's team developed a method that took as an input imagery from the Landsat 8 and Sentinel 1 satellites, on which they trained a convolutional neural network. They pre-trained the CNN on a transfer learning dataset from ImageNet to increase the accuracy of the CNN (compared to simply training on random initializations). The CNN performed well, classifying electricity and sewage infrastructure above 85%, roads at 78%, and piped water at 73% accuracy on LandSat 8 imagery (Oshri *et al*. 2018). This project demonstrates not only the feasibility of feature detection and extraction from satellite imagery with CNNs but also the social benefit these applications can confer on tasks that are necessary but traditionally resource-intensive.

A similar application of CNNs for the monitoring of infrastructure was developed following Hurricane Harvey (Cao and Choe 2018). Just as Ermon and his team recognized the necessity of monitoring infrastructure development in sub-Saharan Africa, so too did two researchers at the University of Washington recognize that timely damage assessment after hurricanes is crucial for facilitating the sensible deployment of first responders and resources to affected areas. Again, wanting to circumvent the huge costs of time and labor that the manual execution of a task like this can require, the duo developed a CNN which classified buildings in satellite imagery of affected areas as either "Flooded/Damaged Building" or "Undamaged Building" with 97% accuracy (Cao and Choe 2018:14).

What about application of CNNs in archaeology? As I mentioned earlier, remote sensing and archaeology share a close history. Computer science and archaeology, however, are not so intimately intertwined. Sarah Parcak, perhaps the doyenne of remote sensing in archaeology, uses highly technical methods to adjust various forms of remote sensing data until unknown archaeological features, once invisible to the naked eye, become visible. These methods include applying contrast enhancement to pixels in satellite imagery to prepare imagery for analysis; combining spectral bands to bring invisible features, like vegetation changes, into clearer view; and image thresholding, or specifying which pixel values will remain visible to the user (Parcak 2009:85-96).

Despite such ground-breaking work, Parcak has expressed some rather traditional opinions about the utility of computers for advancing archaeological inquiry. For instance, in her handbook *Satellite Remote Sensing for Archaeology*, she writes that "computers cannot tell if a site or feature is present or not; they just facilitate the display of pixels. It is up to us to determine what those pixels mean" (Parcak 2009: 109). Granted, this paper handbook was published in 2009, and the general knowledge of computer science within the fields of archaeology and remote sensing has grown immensely in the past decade – one needs only turn to the buzz about digital classics or the growing prominence of and excitement surrounding digital panels and talks at archaeological conferences for proof. Nevertheless, true computer vision tasks in archaeology seem to come largely from computer scientists who see archaeology as a field whose proximity to remote sensing lends itself well to experimenting with feature extraction problems. This is not to say that important work is not being done by professional archaeologists – quite the opposite, as I will explore in a moment – but rather that if we can appreciate the impressive work being done by people outside of the academic domain of archaeology, we can also envision how that same vein of work would benefit from the knowledge of scholars within the domain.

Let us consider some of the groups collaborating across disciplinary boundaries. One team of scholars from various disciplines at The Ohio State University sought to identify high circular tombs, or HCTs, in satellite imagery of the Arabian landscape (Schuetter *et al.* 2013). Because there are tens of thousands of HCTs scattered throughout this region, and because those HCTs tell anthropologists valuable information about settlement patterns and tribal dynamics, identifying and mapping them is an important and huge task. The scale of the task is further compounded by the fact that the tombs rather blend in with the surrounding landscape when viewed in satellite imagery, and that the tens of thousands of tombs are scattered over a massive landscape for which a mere 0.2% of the satellite imagery contains pixels belonging to HCTs.

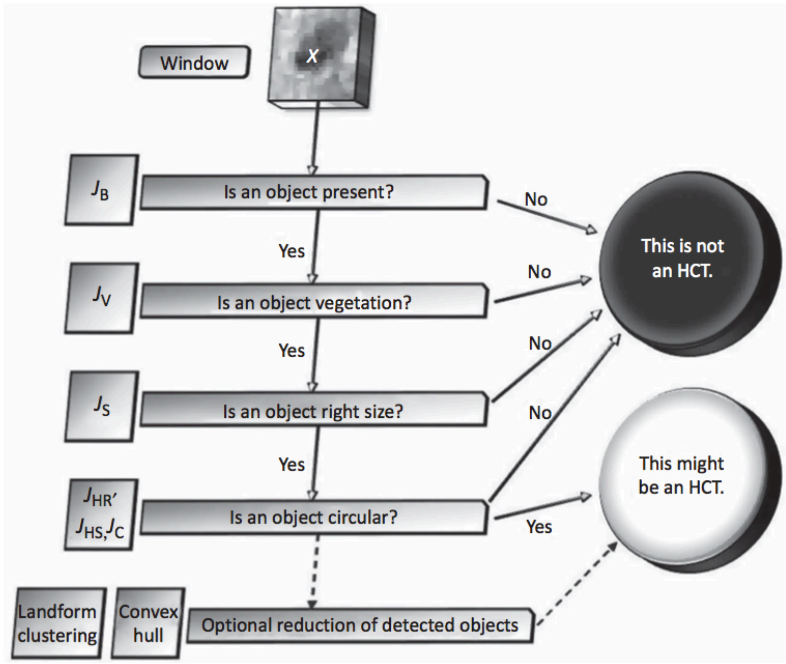

Figure 4. An overview of the steps in the HCT detection algorithm.

Figure 6: overview of computer vision algorithm employed by Schuetter *et al.*2013 in detection of Arabian tombs in satellite imagery.

Although the team did not use a convolutional neural network, their algorithmic approach was elegant and effective (Schuetter *et al.* 2013:6619-6625). Using a series of algorithms like the Canny edge detector, Hough circle fitting, and boundary extraction, the algorithm detected candidate HCTs in the satellite imagery. Even with a total dataset of 76 tombs – and a training set of only 26 – the algorithms still autodetected tombs at above 50% accuracy, reaching as high as 92% accuracy (Schuetter *et al*. 2013).

BACKGROUND ON HISTRIA

My proof-of-concept code centers on extracting tumuli from satellite imagery. Hopefully having thoroughly demonstrated the efficacy of convolutional neural networks for feature of extraction, I will now introduce the area that is the focus of my code and the archaeological features in this area that I attempt to extract.

The satellite imagery used in my algorithm is taken of and around the ancient site of Histria. Histria is located in present-day Romania along the western coast of the Black Sea, 80km south of the southernmost arm of the Danube, along a peninsula that extends along Lake Sinoé and Lake Histria. The colony takes its name from the Thracian name for the Danube river, *Istros* (Donnellan 2004:201).

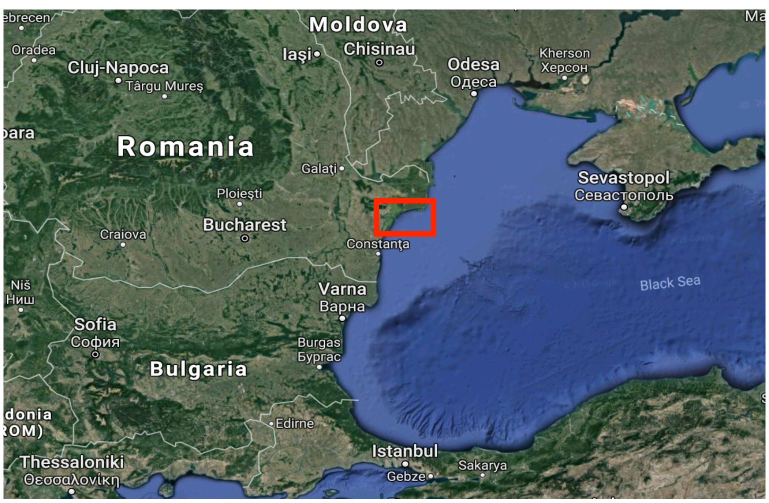

Figure 7: approximate location of Histria on map of larger Black Sea area (Google Earth).

The colony itself is comprised of an acropolis, located on the highest point of the peninsula and overlooking the Romanian countryside and the rivers Nuntasi and Iunan-Dere, and also a necropolis, situated just to the north of Lake Histria and whose surface measures approximately $5km^2$, though the full extent and its boundaries are not exactly clear. Surrounding the city proper is the chora.

Histria was an offshoot of Miletus and the first Milesian colony settled in the Black Sea region. Colonization was a hallmark practice of the Archaic Greeks, (ca. 8th-6th century BC) in large part spurred on by population growth in this period that outpaced their ability to exploit the natural resources of their homeland. The Greeks sent out colonies all around the Mediterranean basin, expanding from Attica and the Aegean islands westward to Spain, eastward to the Levant and southward to the coast of Africa. Scholars estimate some 200 colonies were settled around the Black Sea alone (Petropoulos 2003:17), out of the approximately 500 total Greek colonies that, by the start of the 6th century, accounted for 40% of all Greeks (Cartwright 2018). Miletus was particularly fruitful in its colonization of the Black Sea region, and possible explanations for the Milesians' prodigious expansion include a desire to establish a "North-east passage" to bring oriental bronzes from Armenia into the Greek world, a hunger for more land, and the promise of strategic trading posts in the region (Boardman 1999:239-243).

The exact foundation date of Histria is debated, but scholars agree that Milesians had settled there by at least the middle of the seventh century BC, and by 630 at the latest (Petropoulos 2003:26). In any case, the archaeological record indicates that Histria was well developed by the turn of the century (Donnellan 2004:204-205), and, having minted its first coins in 480 BC, was commercially active at the close of the Archaic period (Andrews 2010:55).

The first excavations at Histria took place under Vasile Pârvan in 1914, illuminating artifacts from the Roman period (ca. 3rd-5th c. AD). Following Pârvan's death in 1927, excavation continued until 1941 under the direction of Scarlat Lambrino. Unfortunately, most of Lambrino's notes and research on the region never saw publication, and attempts by subsequent directors to recover the knowledge obtained by Lambrino during excavation were unsuccessful. Excavations resumed in 1950 under the direction of Emil Condurachi, who, for the first time, dug beneath the Roman layers at the site and began to uncover the earliest layers of Greek settlement.

Condurachi was also the first to excavate Histria's necropolis in 1955. Though untouched by any of Condurachi's predecessors, the necropolis was nevertheless evident to them and, indeed, to all since antiquity, from the more than 1,000 tumuli visible to the naked eye. The population at Histria buried their dead in these tumuli (singular: tumulus), or burial mounds. In the introductory chapter of a large volume on tumuli in antiquity titled *Tumulus as Sema* (Henry and Kelp 2016), Susan A. writes that a tumulus is

> most basically, a bump on the ground … [that] may have been constructed to contain a tomb (of varying qualities of construction and elaboration), but it might also be an empty artificial mound. Or it might originally have been an artifact of cultivation, of rock clearance and of ploughing. Or it might even be a 'mere' natural hillock (Alcock 2016:1).

Additionally, although tumuli are most popular in the Black Sea region and Mediterranean, they are also a global phenomenon, cropping up across Europe, in East Asia, and in South America (Alcock 2016).

Tumuli at Histria are concentrated in the colony's necropolis, which saw a period of use from the mid-6th century BCE to the 2nd century CE. Only 34 of the excavated tumuli have been published, and a number of features – like their construction, associated finds, and topography – have led archaeologists to delineate the phases of the necropolis into three basic groups. The first phase dates from the mid-6th century until the mid-4th and is associated with 14 of the 34 tumuli; the second phase dates from the mid-4th until the end of the 1st century BC and claims 13 of the tumuli, and the third phase dates to the Roman period and claims the remaining seven (Donnellan 2004:204-205).

Tumuli burials at Histria differ from tumuli burials at other centers in the Black Sea region on two major accounts. First, the tumuli are constructed purely of earth and do not contain an interior chamber constructed of wood or stone. Second, the majority of tumuli – 32 of the published 34 – contained cremation burials rather than inhumations, a peculiarity of Histria compared to the rest of the region, whose burials rituals predominantly favored inhumation.

More can be said about the tumuli with regards to the construction and the ritual of burial. In the first case, three major construction types emerge: those with a circular ditch constructed around the periphery of the tumulus, belonging to the oldest phase of the necropolis; those containing an internal funerary platform, measuring from 0.3 to 0.5 meters in height; and those containing a stone circle rather than a funerary platform, serving the same ostensible purpose of indicating the place of burial but present in only two of the published tumuli.

As for burial ritual, again we can identify three types: tumuli with cremation occurring at the same place as the burning (primary cremation), tumuli with cremation occurring elsewhere from the burning (secondary cremation), and inhumation. Archaeologist Petre Alexandrescu even further subdivides the tumuli containing cremations into nine subgroups based on the shape and depth of their associated cremation pits, though these fine distinctions do not bear repeating here. It is worth noting, though, that of the 34 published tumuli, of which 32 contained cremations, only three contained secondary cremations. This further distinguishes Histria from other colonies in the Black Sea region. Though cremation graves have been discovered in every necropolis on the Bulgarian coast of the Black Sea and at others in the circumpontic region, the rite remains relatively rare.

Some understanding of the ethnic makeup of the Histrian population can help explain these burial practices that seem to be peculiar to Histria. Greeks and Thracians composed the two main ethnic groups present at Histria, Thracians being those people indigenous to the region extending over most of the Balkans and down the western coast of the Black Sea, into the Bosporus straight to the east and northern Greece to the west. In particular, those Thracians indigenous to the Histrian region were called Getae, and the Getae and their ancestors occupied the Histrian chora well before the arrival of the Milesians. Following the arrival of the Greeks in the area, however, material culture from settlements in the Histrian chora indicates both Thracian and Greek presence in those areas, indicating the Greek population engaged outwardly with the native Getans. Additionally, archaeological evidence in the form of ceramics implies Getan presence in the Histrian city proper in even the city's earliest period. Much of the traditional research on the area indicates an intertwined but nevertheless distinct relationship between the Greeks and Thracians at Histria, forgoing a more nuanced understanding of ethnicity at the site. Most notably, the population at Histria was not comprised only of Milesians and Getans, but also likely of Greeks from other poleis, as well as Scythians. It seems impossible that all of these groups lived completely distinct lives, and we should assume that the different groups living at Histria adopted some cultural practices from each other in a process that post-colonial studies would deem *creolization* or *hybridization* (Donnellan 2004).

To return to the tumuli, this process of hybridization has major implications for the burial practices at Histria. While some of the excavated tumuli fit squarely into purely Greek or Thracian burial types, many tumuli contain, in one grave, elements thought to belong to multiple funerary traditions. Even in the earliest tumuli we see deviations from traditional Greek and Thracian grave types, indicating that the process of hybridization had begun to occur already. Fast forward a few generations at Histria and the features of these tumuli begin to converge upon a more similar type, no longer belonging to distinctly Greek or Thracian practices but, instead, a hybrid of local and Histrian traditions (Donnellan 2004).

Further excavation and publishing of the tumuli and graves at Histria will be integral for fully understanding the identity of the peoples present at the site, though the question of identity is not within the scope of this paper. Still more can be told about the population at this site, though, by examining the distribution of tumuli in Histria proper and throughout its chora. However, a systematic mapping of the thousands of tumuli in the Romanian countryside is incredibly resource intensive. This might take the form of an archaeological survey – requiring lots of manpower – or an examination of satellite imagery, requiring less manpower, but an excruciating attention to detail and nontrivial investment of time. Training a computer to identify and extract tumuli from satellite imagery of the relevant landscapes, then, promises to reduce the time and manpower involved in a task of this scope.

As I mentioned in my section on machine learning, this project uses supervised machine learning to perform a classification task. Below is a visualization of supervised classification.

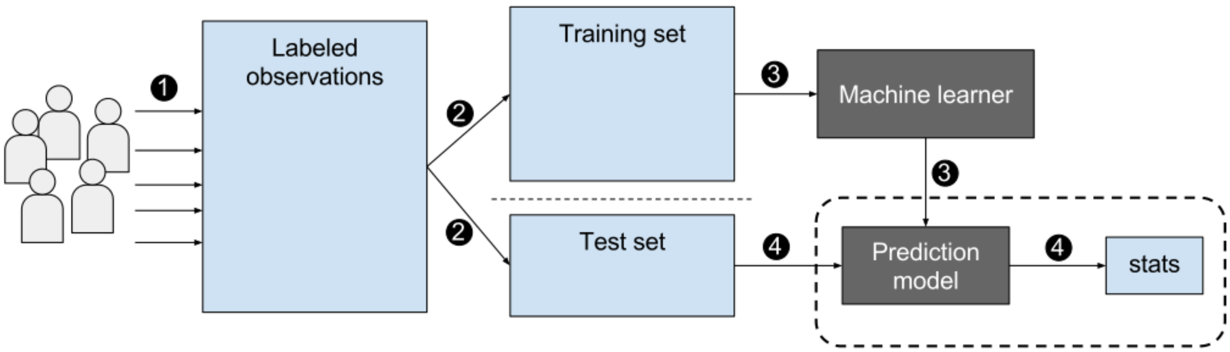

Figure 8: Diagram of Machine Learning Phases (after Salian 2018).

As the graphic of Figure 8 indicates, a set of images and labels are fed into a machine learning algorithm – in my case, a convolutional neural network. For my project, the images are from satellite imagery of Histria, Romania, and the labels are bounding boxes indicating the presence of tumuli. Figures 9 and 10 offer an example of unlabeled imagery and that same imagery labeled with bounding boxes.

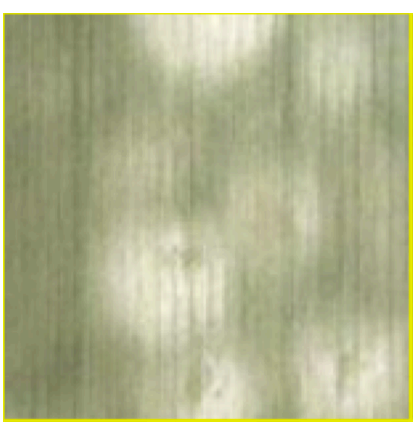

Figure 9: unlabeled satellite imagery

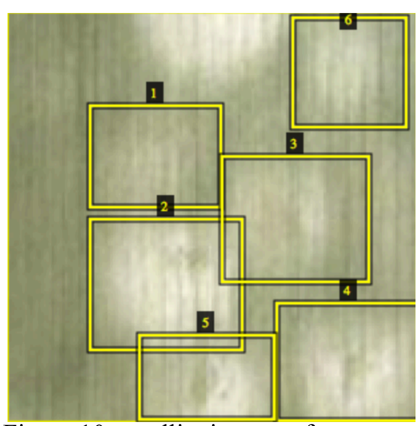

Figure 10: satellite imagery from fig. 9, labeled with bounding boxes

The CNN extracts features from the imagery – the salient features in my case being the tumuli – and the machine learning algorithm then trains on these features and labels until it has seen the entirety of the training set. At this point, the machine learning algorithm has some idea of what the labeled object "looks like" (though it might not be terribly accurate, – see figure 11). In this case, we hope that the algorithm thinks tumuli are approximate circles a few dozen pixels in diameter that appear lighter or darker in color than the surrounding landscape.

After the algorithm "learns" on the training set, one then supplies it with a new set of unlabeled images that it has never seen before. The hope, now, is that the machine has accurately learned to recognize the feature you trained it to identify, and that the machine can accurately detect those same learned features in a fresh set of images. For an object detection problem like the one in my code, the algorithm's output is labels that it produced itself, in the form of bounding boxes, on images it has never seen before.

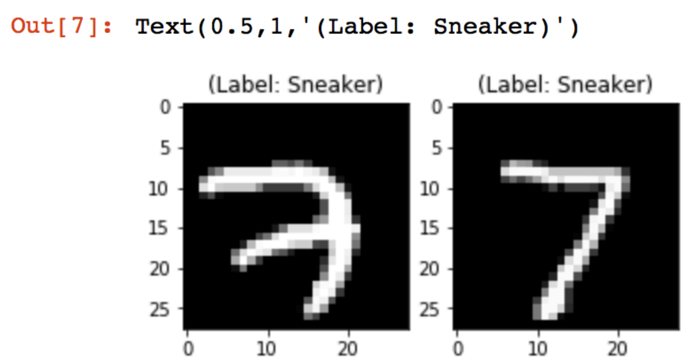

Figure 11: Example of incorrect output data. I accidentally used images of handwritten numbers as the testing set on a classifier that I trained on clothing items. The classifier thinks the number 7 is a sneaker.

In this section, I will now describe my methodology, from collecting and labeling data to building the convolutional neural network. Interested readers can find this code and follow my progress at https://github.com/mallorywillett/thesis-CNN.

A. Data description

1) The raw imagery is Bing aerial imagery, which draws from DigitalGlobe data sources
2) I uploaded the raw imagery into QGIS, an opensource geographic information system, and overlaid a grid layer on the raw imagery to help visualize squares of equal size in the landscape
3) I took screenshots of squares of uniform size (155x155 pixels each) in the grid with my native Mac screenshot tool, noting the coordinates of the squares in order to keep track of where each image came from in space
4) I labeled my images with the VGG Image Annotator (http://www.robots.ox.ac.uk/~vgg/software/via/)[3]
5) I split my labeled images on an 80/20% split into training and testing sets, respectively

B. Labels

After labeling my images using the VGG Image Annotator, I exported the labels/annotations. The VGG Image Annotator exports the labels as a JSON file. JSON stands for JavaScript Open Notation, and a JSON file is a kind of text file that represents data in a way that is easy for humans and machines to interpret because it is stored as key/value pairs in a dictionary. Because I have trouble easily interpreting JSON files, I wrote a script to extract the necessary information

[3] I initially chose this labeling tool because it allowed me to draw circles on my imagery, to most closely capture the tumuli, whereas many labeling tools only support polygon annotation. I later realized that the TensorFlow object detection API that I employed in my CNN used bounding boxes (squares) to detect objects, and so ended up labeling my tumuli with squares rather than circles. In the future, I would use the LabelImg annotation tool (https://github.com/tzutalin/labelImg) to label my data, because the annotations are saved as XML files in PASCAL VOC format. The PASCAL VOC format is one of the record formats supported by TensorFlow's object detection API, so exporting annotations in this format would save you the step of writing a custom script to transform your annotations into a format that TensorFlow can work with (as I had to do with my outputs from the VGG annotator).

from that JSON file and display in plain English text the x and y coordinates of the top left corner of each bounding box in a given image, along with the width and height of those bounding boxes (this step is not strictly necessary).[4] I then split my labeled data into a training set, containing 80% of the labeled images, and a testing set, comprising the remaining 20% of the images, whose labels I withheld from the CNN.

C. Object Detection

I elected to use the TensorFlow Object Detection API[5] for my CNN. TensorFlow is an opensource machine learning library that provides the backbone for the machine learning performed by major companies like Google, Twitter, and Intel (https://www.tensorflow.org/about?). The most popular machine learning framework by most metrics (Hale 2018), TensorFlow is well documented and has a massive community of users, meaning resources and solutions to common problems abound. This makes TensorFlow a powerful and friendly framework for newcomers to deep learning like myself.

TensorFlow's Object Detection API use a special file format called TFRecords. This is not the format that my original labeled data takes, so I used the script detailed on TensorFlow's GitHub page[6] to convert my data into the correct file format.

---

[4] This script, and the whole of my code, is available on my GitHub.
[5] API stands for Application Programming Interface. An object detection API is basically a set of prewritten commands that simplify the task of creating an object detection model, essentially making it so that you don't have to reinvent the wheel with a new script for every new object detection task. Instead, you can use prewritten functions built into the API.
[6] https://github.com/tensorflow/models/blob/master/research/object_detection/g3doc/using_your_own_dataset.md

D. Training the model

Having my labeled images in the right format, I could then begin training. For this, I closely followed the instructions of a blog post called "How to train your own Object Detector with TensorFlow's Object Detection API" (Tran 2017).

To train the model, you need the dataset of TFRecord files and a corresponding label map. Because my object detection model was only looking for one class of object – tumulus – my label map looked like this:

```
# Create dictionary of target classes
label_dict = {
        1: 'tumulus',
}
```

The next step in training the model is building an object detection pipeline, which I have not yet completed. This pipeline is where you identify parameters for the neural network like batch size, training iterations, and learning rate – variables which I will adjust with as I grow my dataset and assess the model's accuracy

Because my set of training images is currently small, I intend to train my model on my own machine[7]. As I scale the project upwards to include more imagery of the Histrian chora, however, I will likely take a cloud-based training approach in order to free up my own machine by harnessing the power of more powerful computers. Services like this are offered through Amazon Web Services and the Texas Advanced Computing Center.

[7] Rather than dedicate too much time to labeling data, I decided instead to only label a small set of 20 images, in order that I could dedicate more time to writing the CNN itself. After my CNN is up and running, I can then go back and add more labeled data to increase the accuracy of the model, which will require only a trivial adjustment to the existing code.

REFLECTION AND LESSONS FOR CONTINUATION

Early in this project, I realized that this was a task far larger than I could handle alone in the given amount of time. Although previous to beginning this code I had completed eighteen hours of coursework in computer science, including formal education and coding experience in machine learning that gave me the boldness to undertake this project in the first place, I nevertheless am not a computer science major, and my experience with convolutional neural networks was extremely limited before undertaking this project. Without a faculty mentor in the computer science department, I found myself struggling to overcome even small roadblocks due simply to the fact that I did not know where to look for the answers to my problems.[8]

I learned immediately, therefore, the value of collaboration in academia. In the fall semester, a time dedicated to gathering sources before the big writing push in the spring, I found myself arranging perhaps a dozen meetings with academics and students in various fields for advice on approaching my problem. It was all incredibly valuable and pushed me closer and closer to my goal. Still, writing the code was an enormous hurdle, and having a team of people help me with it would have made the task far more manageable and the outcome far more successful. Ultimately, archaeologists wishing to implement a CNN for feature extraction would do well to team up with a computer scientist with CNN/deep learning experience. As I continue the work on this code in the future, I will undoubtedly lean on those with more neural network expertise than myself.

This process was essentially eight long months of trial and error, some errors being more painful than others. While every iteration of trial and error improved my workflow and taught me more about how to solve my problem, each of these iterations simply lasted too long compared

[8] The author was surprised to learn that the body of knowledge on Stack Overflow does, indeed, have its limits.

to the total possible duration of the project. In a particularly painful instance of this, I screenshotted and renamed dozens of images and labeled hundreds of instances of tumuli only to realize that the images had to be of identical dimensions (mine, naturally, were not).

I thought that I should approach the problem in steps, and that step one was to obtain and label all of my data. Often, this is how we approach problems in the humanities and archaeology. First we trace the soil change laterally to uncover its visible extent, then we work downward, removing and recording the entirety of the context. Or perhaps we want to write a big paper, and we begin by dedicating a large chunk of time to researching and gathering sources before putting any words on a page. But software development embraces an approach called agile development. At its core, agile development embraces continual dialogue among developers and between developers and clients to "efficiently and effectively respond to user requirement changes" and prevent the "substantial financial loss" that can arise from traditional, non-agile development practices (Lee and Xia 2010:88).

After spending September through February on a non-agile development approach with very little measurable progress on my code, I adopted the agile method as detailed in the book *Sprint*, which focuses on short feedback loops and building facades of prototypes that are just real enough to test but not so real that they take weeks or months to develop (Knap *et al.* 2016:166). This meant a major change to my method: instead of labeling all of my images in one go and building up the code from there, I decided to write a complete code that worked at the smallest, most basic level possible. Because I was not spending hours labeling hundreds of images, I was more willing to test different object detection APIs and frameworks, even if this meant I had to re-label my data – because now I only had a small handful of labels to fix in the first place. As I continue to develop and debug my algorithm, this agile approach allows me to focus more of my time on the most essential parts of my code. And when my CNN is finished, I can be confident that I chose the best approach for this research, instead of simply sticking to a method because I had spent too long working on it to let it go.

CONCLUSION

Although the results of my proof-of-concept are still forthcoming, the efficacy of convolutional neural networks for object detection and feature extraction is unquestionable. Utilizing CNNs in the fields of cultural resource management and archaeology promises not only to decrease the human labor involved in manual feature identification but also to better inform archaeologists and conservationists of the status of cultural heritage sites, leaving experts better equipped to protect those sites. At Histria, locating unmapped tumuli might protect them from destruction as the modern city expands and sea levels rise; globally, sites already known to archaeologists stand to be more closely monitored with this method, and still more sites stand to be discovered. A collaborative effort between computer scientists and archaeologists combined with a thoughtful application of convolutional neural networks in the field of archaeology shows great promise for advancing the goals of cultural resource management.

Works Cited

Abolt, Charles J., *et al.* "Brief communication: Rapid machine-learning-based extraction and measurement of ice wedge polygons in high-resolution digital elevation models." *The Cryosphere*, vol. 13, 2019, pp. 237-245.

"About the GlobalXplorer project."*GlobalXplorer*, 2018, www.globalxplorer.org/about.

Agapiou, Athos, *et al*. "Impact of urban sprawl to cultural heritage monuments: The case study of Paphos area in Cyprus." *Journal of Cultural Heritage*, vol. 16, 2015, pp. 671-680.

Albert, Adrian, *et al.* "Using convolutional neural networks and satellite imagery to identify patterns in urban environments at a large scale." 2017, arXiv: 1704.02965v2 [cs.CV].

Alcock, Susan. "Time Traveling Tumuli: The many lives of bumps on the ground," in Kelp and Henry (Eds.), *Tumulus as Sema: Space, Politics, Culture and Religion in the First Millenium BC*. De Gruyter, 2016, pp. 1-8.

Anandhalli, Mallikarjun and Vishwanath P. Baligar. "A novel approach in real-time vehicle detection and tracking using Raspberry Pi." *Alexandria Engineering Journal*, vol. 57, iss. 3, 2018, pp. 1597-1607.

Anderson, David G. *et al.* "Sea-level rise and archaeological site destruction: An example from the southeastern United States using DINAA (Digital Index of North American Archaeology)." *PLoS ONE*, vol. 12, iss. 11, 2017, pp. 1-25.

Andrews, Smaranda. "Greek cities on the western coast of the Black Sea: Orgame, Histria, Tomis, and Kallatis (7th to 1st century BCE)." PhD Dissertation, Iowa State University. 2010. lib.dr.iastate.edu/etd/11712.

Averett, Erin Walcek *et al. Mobilizing the Past for a Digital Future: The Potential of Digital Archaeology*. The Digital Press @ The University of North Dakota, 2016.

Babawuro, Usman and Zou Beiji. "Automated Extraction of Geospatial Features from Satellite Imagery: Computer Vision Oriented Plan Surveying." *International Journal of Computer Science Issues,* vol. 9 iss. 6 no. 2, 2012, ISSN: 1694-0814.

Banning, E.B., *et al*. "Quality Assurance in Archaeological Survey." *Journal of Archaeological Method and Theory*, vol. 24, iss. 2, 2017, pp. 466-488.

Boardman, John. *The Greeks Overseas: Their Early Colonies and Trade*. 4th ed. London, Thames & Hudson Ltd, 1999.

Brownlee, Jason. "Supervised and Unsupervised Machine Learning Algorithms." *Machine Learning Mastery*, 16 March 2016. www.machinelearningmastery.com/supervised-and-unsupervised-machine-learning-algorithms/.

Burkov, Andriy. "The Hundred-Page Machine Learning Book." Andriy Burkov, 2019.

Cao, Quoc Dung and Youngjun Choe. "Detecting Damaged Buildings on Post-Hurricane Satellite Imagery Based on Customized Convolutional Neural Networks." 2018, arXiv:1807.01688 [cs].

Capper, J.E. "Photographs of Stonehenge, as seen from a War Balloon." *Archaeologia*, vol. 60, 1907, p. 571.

Castle, Nikki. "What is Semi-Supervised Learning?" *Oracle + Datascience.com*, 21 Dec 2017, www.datascience.com/blog/genetic-variants-ai.

Caraher, William. "Slow Archaeology: Technology, Efficiency, and Archaeological Work," in Averett *et al.* (Eds.), *Mobilizing the Past for a Digital Future: The Potential of Digital Archaeology.* The Digital Press @ The University of North Dakota, 2016, pp. 421-441.

Cartwright, Mark. “Greek Colonization.” *Ancient History Encyclopedia.* Ancient History Encyclopedia, 7 May 2018.

Cleere, Christopher, *et al.* “Condition recording for the conservation and management of large, open-air sites: a pilot project at Chersonesos (Crimea, Ukraine).” *Conservation and Management of Archaeological Sites,* vol. 8, iss. 1, 2013, pp. 3-16.

Coifman, Benjamin, *et al.* “A real-time computer vision system for vehicle tracking and traffic surveillance.” *Transportation Research Part C: Emerging Technologies*, vol. 6, iss. 4, 1998, pp. 271-288.

Copeland, Michael. “What’s the Difference Between Artificial Intelligence, Machine Learning, and Deep Learning?” *NVIDIA,* 29 July 2016, blogs.nvidia.com/blog/2016/07/29/whats-difference-artificial-intelligence-machine-learning-deep-learning-ai/.

De Laet, V. *et al.* “Methods for the extraction of archaeological features from very high-resolution Ikonos-2 remote sensing imagery, Hisar (Southwest Turkey).” *Journal of Archaeological Science,* vol. 34, 2006, 830-841.

Donnellan, Lieve. “Ethnic identity in the Western Black Sea area. The cases of Histria, Kallatis and Apollonia Pontika (7^{th}-4^{th}/3^{rd} centuries BC).” *Talanta: Proceedings of the Dutch Archaeological and Historical Society*, vol. 36-37, 2004-2005, pp. 189-267.

Fabiani, Michelle D. “Disentangling Strategic and Opportunistic Looting: The Relationship Between Antiquities Looting and Armed Conflict in Egypt.” *Arts*, vol. 7, no. 22, 2018; doi:10.3390/arts7020022

Grossi, Carolota M., *et al*. “Predicting long term freeze-thaw risks on Europe built heritage and archaeological sites in a changing climate.” *Science of the Total Environment*, vol. 377, iss. 2-3, 2007, pp. 273-281.

Hale, Jeff. “Deep Learning Framework Power Scores 2018.” *Medium*, 19 Sept 2018, towardsdatascience.com/deep-learning-framework-power-scores-2018-23607ddf297a.

He, Kaiming, *et al*. “Mask R-CNN.” The IEEE International Conference on Computer Vision (ICCV), 2017, pp. 2961-2969.

Johnson, Khari. “Alexa AI scientists reduce speech recognition errors up to 22% with semi-supervised learning.” *VentureBeat*, 4 April 2019, venturebeat.com/2019/04/04/alexa-ai-scientists-reduce-speech-recognition-errors-up-to-22-with-semi-supervised-learning/.

Kelp, Ute and Olivier Hendry (Eds.), *Tumulus as Sema: Space, Politics Culture and Regligion in the First Millenium BC*. De Gruyter, 2016.

King, Thomas F. (Ed.), *A Companion to Cultural Resource Management.* Wiley-Blackwell, 2011.

Knapp, Jake, *et al. Sprint: How to solve big problems and test new ideas in just five days.* New York, Simon & Schuster, 2016.

Krizhevsky, Alex, *et al.* “ImageNet Classification with Deep Convolutional Neural Networks.” *Communications of the ACM*, vol. 60, no. 6, 2017,
pp. 84-90.

“LAND INFO High-Resolution Global Satellite Imagery.” *LAND INFO Worldwide Mapping, LLC.* 2018. http://www.landinfo.com/products_satellite.htm

Lee, Gwanhoo and Weidong Xia. “Toward Agile: An Integrated Analysis of Quantitative and Qualitative Field Data on Software Development Agility.” *MIS Quarterly*, vol. 34, no.1, 2010, pp. 84-114.

Lin, Albert Yu-Min, *et al*. “Crowdsourcing the Unknown: The Satellite Search for Genghis Khan.” *PLoS ONE,* vol. 9, no. 12, 2014, e114046. doi:10.1371/journal.pone. 0114046.

MacDonald, Robert. “Corona: success for space reconnaissance, a look into the cold war, and a revolution for intelligence.” *Photogrammetric Engineering and Remote Sensing*, vol. 61, iss. 6, 1995, pp. 689-720.

Marzeion, Ben and Anders Levermann. “Loss of cultural world heritage and currently inhabited places to sea-level rise.” *Environmental Research Letters,* vol. 9, no. 3, 2014, 034001.

Matthies, Larry, *et al.* “Computer Vision on Mars.” *International Journal of Computer Vision*, vol. 75, iss. 1, 2007, pp. 67-92.

Oshri, Barak, *et al*. “Infrastructure Quality Assessment in Africa Using Satellite Imagery and Deep Learning.” 2018. arXiv: 1806.00894 [cs.CV].

Panayotova, Krystina. “Burial and Post-Burial Rites in the Necropoleis of the Greek Colonies.” In *Ancient Greek Colonies of the Black Sea 2,*ed. D.V. Grammenos and E.K. Petropoulos. Vol. 1, Oxford: Archaeopress, 2007, pp. 85-126. Print.

Parcak, Sarah H. “Satellite Remote Sensing for Archaeology.” Routledge, 2009.

Petropoulos, Elias K. “Problems in the History and Archaeology of the Greek Colonization of the Black Sea.” In *Ancient Greek Colonies in the Black Sea*, ed. D.V. Grammenos and E.K. Petropoulos. Vol. 1, Thessaloniki: AIBE, 2003, pp. 17-94. Print.

Proulx, Blythe Bowman. “Archaeological Site Looting in “Glocal” Perspective: Nature, Scope, and Frequency.” *American Journal of Archaeology,* vol. 117, no. 1, 2013, pp. 111-125.

Salian, Isha. “SuperVize Me: What’s the Difference Between Supervised, Unsupervised, Semi-Supervised, and Reinforcement Learning?” *NVIDIA,* 2 August 2018. www.blogs.nvidia.com/blog/2018/08/02/supervised-unsupervised-learning/.

Schuetter, Jared*, et al*. “Autodetection of ancient Arabian tombs in high-resolution satellite imagery.” *International Journal of Remote Sensing*, vol. 34, no. 19, 2013, pp. 6611-6635.

Silver, David, *et al.* “Master Chess and Shogi by Self-Play with a General Reinforcement Learning Algorithm.” 2017, arXiv: 1712.01815v1 [cs.AI].

Taigman, Yaniv, *et al*. “DeepFace: Closing the Gap to Human-Level Performance in Face Verification.” *Conference on Computer Vision and Pattern Recognition,* Columbus, OH, 24 Jun 2014.

Taubman, Philip. *Secret Empire: Eisenhower, the CIA, and the Hidden Story of America's Space Espionage.* Simon & Schuster, 2003.

White, Gregory G. and Thomas F. King. *The Archaeological Survey Manual.* Walnut Creek, Left Coast Press, 2007.

Yao, Mariya. “Chihuahua or muffin? My search for the best computer vision API.” *Medium*, 12 Oct 2017, medium.freecodecamp.org/chihuahua-or-muffin-my-search-for-the-best-computer-vision-api-cbda4d6b425d.

BIOGRAPHY

Mallory Willett was born and raised in Coppell, Texas. She enrolled in the Plan II Honors program at the University of Texas at Austin in 2014, also completing a Bachelor of Arts in Ancient History and Classical Civilization in the Classics department and an eighteen-hour Elements of Computing certificate in the Computer Science department. In her senior year, she presented a talk on her thesis at the joint annual meeting of the Archaeological Institute of America and the Society of Classical Studies, and was published as co-author for her statistical work on a study in the *Journal of the American Medical Informatics Association.* She graduated Phi Beta Kappa in 2019 and will begin work as a technical solutions engineer at Epic Systems in Madison, WI in the summer.