

Universiteit
Leiden
The Netherlands

# Deep learning solutions for domain-specific image segmentation

Vadineanu, S.

**Citation**
Vadineanu, S. (2025, October 8). *Deep learning solutions for domain-specific image segmentation*. Retrieved from https://hdl.handle.net/1887/4266937

| | |
|---|---|
| Version: | Publisher's Version |
| License: | Licence agreement concerning inclusion of doctoral thesis in the Institutional Repository of the University of Leiden |
| Downloaded from: | https://hdl.handle.net/1887/4266937 |

**Note:** To cite this publication please use the final published version (if applicable).

# Chapter 5

# Explainability and Annotations with Activation Maps

## 5.1 Introduction

Automating archaeological feature detection and classification on remotely sensed imagery is increasingly becoming possible. Until recently, the reliability of object-based solutions, i.e., the partitioning of remote-sensing images into categories [37], suffered from the sensitivity of algorithms to image variations, e.g., in contrast, or brightness, or from the heterogeneity of archaeological objects as algorithms expect homogenous entities [88]. While these methods were once considered improvements over pixel-based classification (as early examples, see [31, 33]), object-based methods [40] could not be fully used for automatically detecting the relevant archaeological features of remote-sensing images.

However, a new approach is forming thanks to the fast-emerging deep learning

This chapter is based on:

> Ș. Vădineanu, T. Kalayci, D. M. Pelt, and K.J. Batenburg. "Convolutional Neural Networks and Their Activations: An Exploratory Case Study on Mounded Settlements". *Journal of Computer Applications in Archaeology, 7(1).* ubiquity press (2024).

paradigm that intends to bypass earlier obstacles by learning definitory patterns of the target objects directly from the data. Due to recent advancements in hardware technology and the availability of abundant data, machine learning, especially deep learning, algorithms have seen widespread and rapid adoption in many domains, including archaeology. Deep learning algorithms particularly achieve state-of-the-art performance via convolutional neural networks (CNNs) for many image processing tasks, such as image classification.

In archaeology, Bayesian regularization and Levenberg–Marquardt algorithms have been compared for predicting metrics of Neolithic laminar artefacts [146]. Similarly, machine learning algorithms have been employed to cluster cultural and technological groups within archaeological datasets [147]. Deep learning approaches have also proven successful in detecting and segmenting archaeological structures from LiDAR data [63] and in semi-automatically mapping archaeological topography using airborne laser scanning data [145]. CNNs have facilitated the detection of "princely" tombs [25], and have revealed shell-ring building practices by Archaic Native Americans [38]. Additionally, deep learning-based automated analysis has been applied to archaeo-geophysical images, enhancing the interpretation of geophysical survey data [87].

Nevertheless, the new paradigm already signals it is not devoid of problems such as the requirement of large quantities of high-quality annotated data, high computational costs, and the opacity of the CNNs' decision process. In this chapter, we highlight two of these key issues that might benefit from further research: the annotation cost and the explainability of network architectures. As a constructive approach to address these issues, we present ways to link annotation and explainability problems through visualizations, supported by exploratory statistics.

The annotation problem is particularly relevant to archaeology. While deep learning algorithms are highly effective for numerous imaging tasks, their training demands substantial annotated data. The challenge lies in generating annotations, especially in specialized domains such as archaeological satellite imagery, where annotations are often created by trained experts with limited availability [15, 73]. Annotated data scarcity becomes particularly problematic for labour-intensive tasks, such as segmentation, which requires classifying the pixels within an image. Such constraints can impede the practicality of deep learning applications. Therefore, addressing the challenge of annotated data scarcity can play an important role in the further adoption of deep learning in archaeological research.

We also observe that achieving high accuracy is the main concern in the schol-

arship. When provided with sufficient training data, recent deep learning models generally produce highly accurate detection and classification results regardless of the architecture. Yet, the influence of architectural choices over which image features contribute towards a prediction receives less attention. This perceived opacity of neural networks' decision-making process may contribute to some research fields approaching their use with caution. Therefore, besides alleviating the burden of extensive manual annotation, visualizing what the most relevant image areas are for a given prediction can build trust among practitioners. Moreover, such insights can assist the experts in developing less biased workflows [100, 142], rectifying mis-annotated samples or discovering new patterns in the images.

To address the two key issues outlined above, we utilize explainability techniques, i.e., methods producing visual interpretations of a CNN's output in relation to its input, whose results we refer to as activation maps. Particularly, we focus on the explainability techniques producing activation maps reflecting the contribution of each individual input pixel towards a CNN prediction. To address the annotation scarcity and the perceived opacity of deep learning, we employ the resulting activation maps as sources of both cheap annotations and insights into the patterns found by CNNs. We address the annotation task by proposing an automated annotation pipeline for generating segmentation masks of archaeological sites from the activation maps extracted with explainability techniques. We apply these techniques to trained classification CNNs, whose training annotations are relatively cheap to produce compared to segmentation masks. We also explore to what extent we can extract meaningful visual insights from the features deemed relevant by different types of CNN architectures. We compare the activation maps extracted from multiple network architectures and study which parts of an archaeological feature contribute the most to the network's predictions. Additionally, we verify whether the highlighted features can signal the presence of mis-annotated images or overfitting. Our integrated workflow helps us to explore the annotation and explainability issues in tandem.

In our workflow, we employ Occlusion Maps [57], LayerCAM [73], and Guided GradCAM [129] as explainability methods. To combat the lack of spatial resolution associated with existing techniques, we also propose an extension to Guided GradCAM. As a case study, we map the extent of ancient settlement mounds within CORONA satellite images in the Upper Khabur Basin of Upper Mesopotamia. We apply these four explainability techniques to three widely used CNN architectures: VGG [132], ResNet [66], and DenseNet [71]. Finally, we explore activation maps to localize ancient settlement mounds using CNNs trained for binary image classification by employing

only image-level annotations.

Our aims are twofold: (i) providing an analysis of the visual cues that contribute to CNNs' predictions of sites from remote sensing images of the Upper Khabur Basin and (ii) using visual cues from activation maps as sources for segmentation annotations. To achieve these goals, we utilize existing explainability techniques and we also propose a new method for extracting activations that better match the expert interpretation of a site than existing works.

## 5.2 Background

### 5.2.1 The Study Area

The Upper Khabur Basin is located within the larger gently undulating plain of Upper Mesopotamia that stretches east-west between the massive Anti-Taurus Mountains in the north and the short mountain range called Jebel Sinjar in the south [42]. The Abd-al Aziz mountain ridge rising across Sinjar also bounds the study area. The primary contributor of the hydrological system is the Euphrates River. Running down from the northwest of Lake Van at an approximate altitude of 3,500 meters, the river significantly drops its gradient as it further moves into the Upper Mesopotamian plain, in modern-day Syria. The Khabur Basin (Figure 5.1) takes its name from the Khabur River, the largest tributary to the Euphrates.

In the Upper Khabur, several wadis (Aweij, Khanzir, Jaghjagh, Jarrah, Kuneizir, and Rumeilan) run in north-south direction eventually draining to Wadi-el-Radd [42, p. 173]. Wadi is an Arabic term denoting a valley-like morphological feature that is dry except during periods of rainfall. Even if they were temporal and usually short-lived, flowing water contributed to the geography and life. Therefore, they "played an important role for human societies within this area and many archaeological sites—often tells (settlement mounds)—are located along them." [41, p. 337] (Figure 5.2).

### 5.2.2 Settlement Mounds

The long-term accumulation of everyday-life cultural material through centuries results in a particular site type, called a settlement mound [126]. These are signature settlements in southwest Asia, getting the names of tell in Arabic, tepe or chogha in Farsi, or höyük in Turkish [101]. Yet, it is important also to note that other regions in the world also host mounded settlements, including Greece [43] and Hungary [113]. Depending on their (post-)depositional processes, density and duration of occupation,

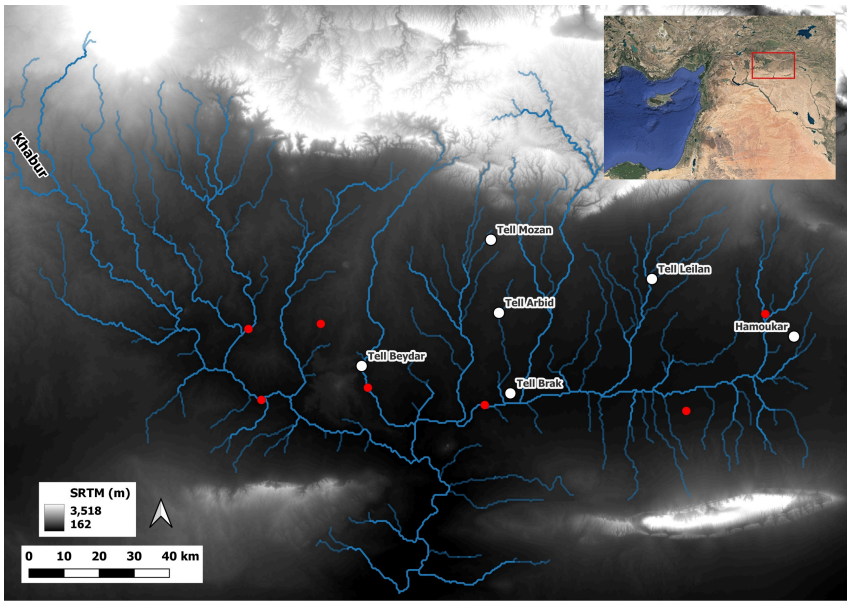



**Figure 5.1:** We investigated mounded sites in the Upper Khabur Basin of Upper Mesopotamia. One can see the 100-kilometre-long Sinjar mountain range in the lower-right corner. The Abd-al Aziz mountain ridge is across the river from Sinjar. Some key settlements in the area are shown in white. Red dots indicate the locations of other settlements discussed in the text.

local geological and geographic conditions, and many other factors mounds exhibit considerable differences. These differences, however, bear the potential for morphological analysis [149, 23]. Mound morphology is almost always variable, but it is possible to identify some broad trends also in our study area. Using the results of Tell Beydar Survey [150] and Tell Hamoukar Survey [148] one may summarize site morphologies, but only briefly and only with great generalization: due to less intense occupation, smaller/lower mounds were formed primarily during early prehistoric times. Rapid nucleation during the second half of the Early Bronze Age (mid-second millennium BCE) resulted in taller and more prominent mounds. From the Late Bronze Age onwards, including the Iron Age, less intensive occupation was attached to the now-abandoned Early Bronze Age mounds. This new phase of nucleation added further to morphological complexities. Lower-density occupation in later periods [97] must have contributed less to the formation of mounds.

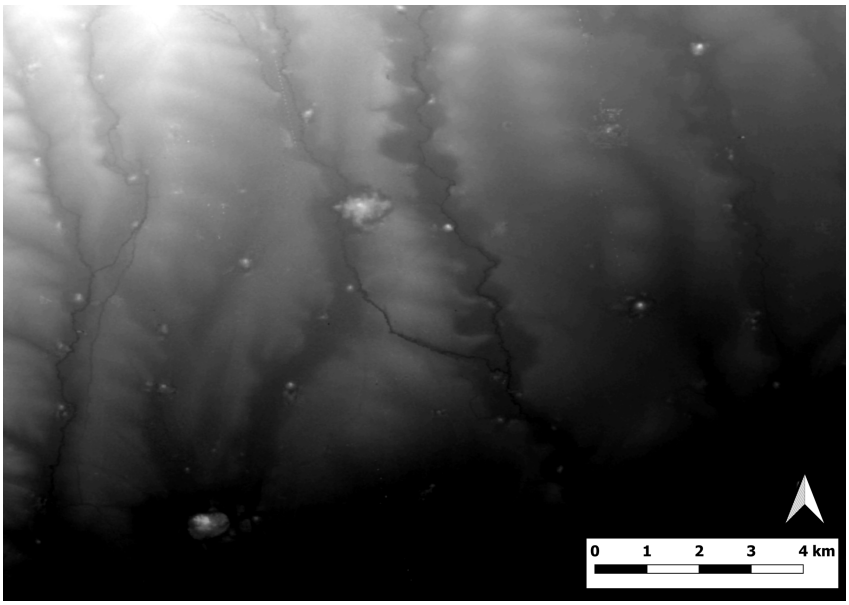



**Figure 5.2:** "Spots" on TanDEM interferometric SAR imagery indicate settlement mounds. They are located mainly along north-south running wadis. The large spot/site at the centre of the image is Tell Sharisi (36.891° N, 41.365° E). In the bottom left, we see another mound, Tell Farfara (36.825° N, 41.334° E).

### 5.2.3 Corona Satellite Imagery

In the study area, tell-sites can be as high as 40 meters and can attain sizes of more than 100 hectares [168]. Due to their considerable sizes and relatively defined site extents, but also thanks to the moderately flat topography of the study area, ancient settlement mounds are visible on remote sensing data, but notably on historical CORONA spy-satellite imagery [119, 24]. In return, it is possible to conduct desktop surveys with visual interpretation [23] and with additional products, such as digital elevation models [103].

The state-of-the-art sensors resolve the ground in great detail and provide data from non-visible portions of the spectrum. CORONA as a historical dataset, but especially the Key Hole KH 4B series (1967–1972) contributes to landscape archaeology in other different ways. At the very least, CORONA predates the negative impact of modern irrigation systems, great dam projects [149, p. 12] and urban sprawl [167, p. 228] on material culture (Figure 5.3).

In particular, the high-resolution of KH-4B (ca. 1.85 meters at Nadir) provides an extensive coverage, mainly due to the panoramic scan. Thanks to multiple CORONA




**Figure 5.3:** State-of-the-art sensors, such as WorldView-2, can resolve the ground in great detail, on the right. CORONA provides historical evidence of land-use land-cover changes, on the left. In this particular example, one can assess the impact of modern buildings on Tell Beydar. Image resolutions are comparable despite the age of spy-satellite imagery.

KH-4B missions, archaeological landscapes can be investigated in time series and the most optimal scenery can be selected for further research. Recent studies highlight the potential of Hexagon [58, 64] and U2 imagery [65]. Yet they are still not widely available for wide-scale analysis. Therefore, ortho-corrected CORONA is still a viable source for exploring diverse archaeological landscapes across the globe.

## 5.3 Deep Learning and Activation Maps

### 5.3.1 Deep Learning for Image Classification

Convolutional neural networks (CNNs) explore patterns in input images through the use of units organized as filters. These filters, forming a convolutional layer, generate intermediary images known as feature maps [92], which essentially represent the prominence of specific features within the image. For example, a filter might emphasize vertical edges, while another filter could identify horizontal edges, textures, and so on. These resulting feature maps then become the input for the subsequent set of filters in the following convolutional layer.

Among the problems tackled with CNNs, we focus on image classification due to its relatively cheap annotation process and widespread relevance. The classification CNN typically comprises two main components: a convolutional part, functioning as a feature extractor, and a fully-connected part, serving as the classifier. In the convolutional part, the learned parameters correspond to the filters within the convolutional layers, while the fully-connected part of the architecture utilizes its learned weights to categorize the features extracted by the convolutional layers. The categorization is performed by reweighing and combining the feature maps in order to produce a set of class probabilities out of which the predicted class is chosen.

Different CNN architectures employ distinct strategies to produce accurate classifications, varying in aspects such as the number of layers, the filter size, and the connectivity between layers. Despite the variety in architectural choices, many CNNs perform similarly well across different tasks and data sets [74]. Moreover, although different architectures may perform similarly on a given task, their inner decision process can vastly differ, thus influencing their explainability and utility as detection tools, therefore, making the selection of suitable architectures a non-trivial problem.

Among the popular well-performing CNNs for the task of image classification, we focus on VGG [132], ResNet [66], and DenseNet [71], listed in the order of their development. All three networks showed particularly good results for the classification of natural images on the ImageNet data set [44], with each network claiming improvements over its predecessor. All three network architectures are still widely used today. Their extensive adoption, architectural differences, and the distinctiveness of remotely sensed imagery from natural images make a comparison between these networks worth exploring. Moreover, such comparison intrinsically contributes to explainability studies by assessing the suitability of the different architectures as visualization tools of relevant patterns within remote sensing archaeological data.

VGG was among the first solutions aiming to improve the classification performance of CNNs by increasing the depth, i.e., the number of layers, of the architecture (Figure 5.4a). This was achieved by reducing the size of the convolution kernels to 3x3, substantially decreasing the number of parameters per layer. VGG consists of several layers where the information is processed sequentially, using the feature maps from the previous layer as input to the next. The feature extractor architecture consists of blocks of 3x3 convolution layers followed by max-pooling layers that reduce the size of the feature maps in half by selecting from every non-overlapping group of 2x2 pixels, the pixel with the highest value. After the final max pooling layer, the resulting feature maps are spatially flattened to sets of 1-dimensional vectors, which

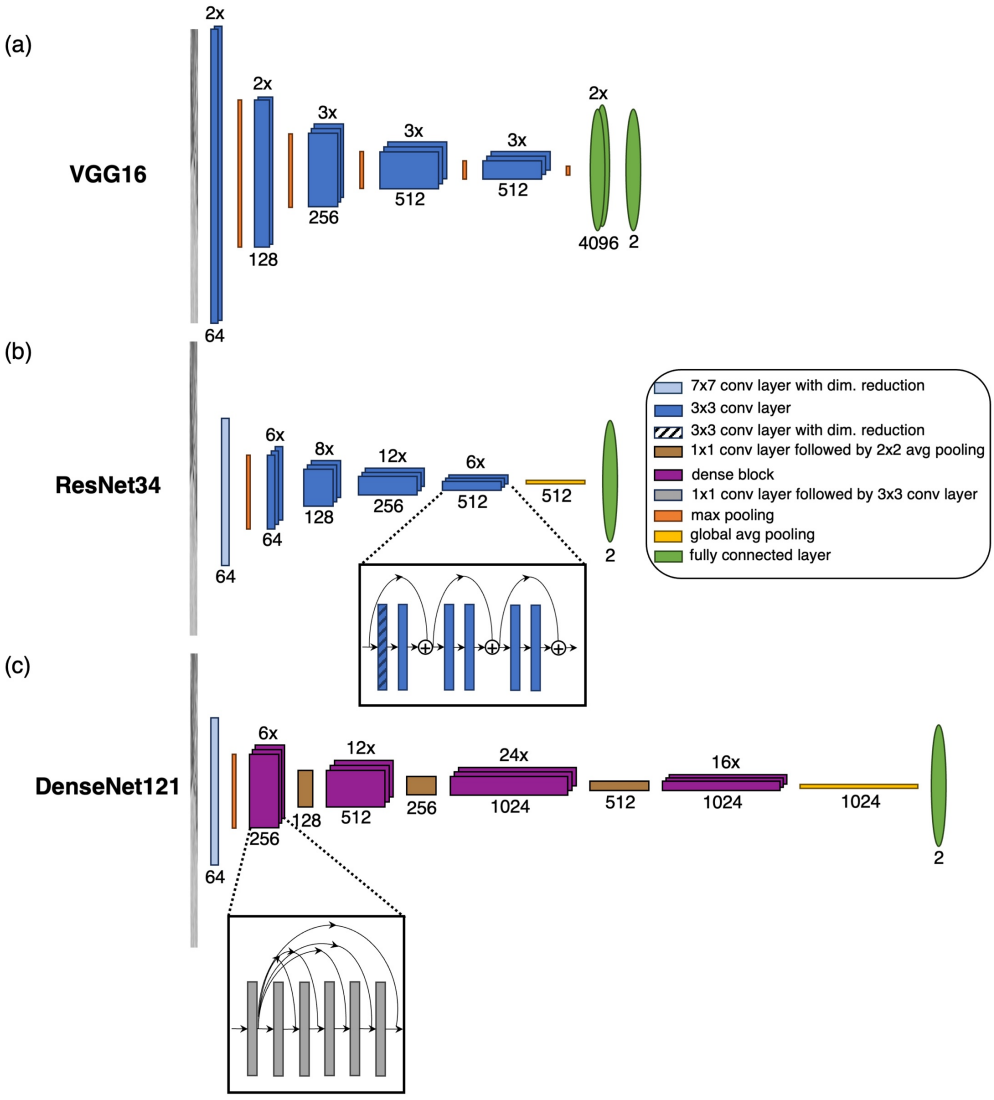



**Figure 5.4:** Schematic representations of VGG16 (a), ResNet34 (b), and DenseNet121 (c). The values above a block of layers correspond to the number of layers within a block. The values below the blocks refer to the number of filters each layer has.

are passed to the classifier.

ResNet is a CNN architecture that focuses on training a larger number of layers than VGG (Figure 5.4b). It achieves this by creating alternative paths that allow for the output of a layer to skip being processed by the immediately following layers.

The unprocessed output is then added to the result of the sequential path. In this way, the network focuses on learning the additions (residuals) that need to be applied to the input such that the relevant features are extracted for classification. This approach allows increasing the number of convolutional layers, enabling the network to learn more complex features. Apart from the alternative paths, ResNet differs from VGG by how it reduces the spatial dimensions of its feature maps since it replaces max-pooling layers with strided convolutions, i.e., instead of sliding the convolutional filter on the image/feature map with a step of one pixel, the step size is increased according to the stride value. After the last set of convolutions, the resulting feature maps are aggregated into a set of 1-dimensional vectors by passing the final feature maps through a global average pooling layer, which reduces a map to the average of the values across all its spatial dimensions. Averaging instead of flattening the final feature maps has the benefit of making the architecture independent of the image size, since the size of the input to the fully connected layer is dictated by the size of the channel dimension of the last convolutional layer, rather than the spatial shape of the feature map after passing the image through the convolutional and pooling layers.

DenseNet also provides a solution for training a large number of layers by developing connections that pass all the feature maps that were created by previous layers to all subsequent layers (Figure 5.4c). The architecture is composed of dense blocks, which aggregate the feature maps from all their convolutional layers, and transition blocks which provide both spatial as well as channel-wise dimensionality reduction. Similar to ResNet, the network makes use of global average pooling to make the transition from the feature extractor to classifier, while relying on average pooling instead of strided convolution for the dimensionality reduction of the feature maps.

There is already a significant number of archaeological case studies using these specific network architectures. However, limited literature is available that considers the rationale behind choosing one architecture type over another. For instance, Albrecht et al. [6] use a VGG to classify archaeological features on LiDAR data. They report they chose VGG because "this approach is accurate and flexible for the archeologist's needs" [6, p. 18]. Similarly, Somrak et al. [136] aim to detect archaeological features on Airborne Laser Scanning (ALS) data using VGG. They used this type of architecture mainly because "[t]here have been previous uses of the VGG network" [136, p. 7]. Verschoof-van der Vaart et al. [154, p. 7] provide more specific reasoning for their choice of the VGG architecture as it "performs better than most shallower networks and needs significantly less memory than some deeper networks, while yielding comparable results.". Patrucco and Setragno [115, p. 19] decided to deploy DenseNet since

"[t]his network allows using fewer channels for each layer, thus having fewer training parameters and a smaller network". Trier, Cowley and Waldeland [144] identified the problem early on. They deploy ResNet18, but also state that "the development of 'general purpose' archaeological CNNs is desirable if the discipline as a whole is to make better use of the methodology." [144, p. 168].

It is also common to use multiple networks and compare results. When multiple networks are used, the major aim is to compare accuracies, leaving little room for advancing research on explainability. Abellán et al. [1, p. 4] use "six architectures to test the accuracy in classifying tooth marks". In another study, researchers worked with seven deep learning models and their choice for the network was based on the ranking of these models [48]. Bonhage et al. [18] further solidify the accuracy problem by asking "what level of accuracy would be required from automated systems to be acceptable for a specific purpose.".

Overall, it appears that when scholars work with a single architecture, there is relatively more discussion on the reasons behind choosing that network. Nevertheless, the rationale behind their choice tends to remain implicit, restricting interpretability. Comparative approaches focus mainly on the accuracy of the results these networks can produce and make limited contribution to our understanding of how different architectures can be exploited to retrieve more information about the data itself. Using explainability techniques can benefit the scholarship as they contribute to understanding whether the image features deemed as relevant by the networks have intuitive explanations.

### 5.3.2 Explainability Techniques

Despite their proven capabilities in increasingly difficult tasks, one major challenge that the current CNNs are facing is a lack of interpretability of their predictions. Consequently, the applicability and reliability of these solutions can be distrusted. In response to this, multiple techniques have been developed to explain the decision process undertaken by CNNs before generating a prediction. In the context of image classification, where the prediction takes the form of class probabilities, these explainability techniques have the added benefit of providing localization information of the most relevant image sections that influenced the prediction of the CNN.

We selected three such techniques, namely Occlusion Maps (OM) [57], Gradient-Weighted Class Activation Maps (GradCAM) [129], and LayerCAM [73]. In our work, we also propose a localization technique based on Guided GradCAM [129]. All the

selected methods have the benefit of being independent of the type of CNN being used, offering good flexibility for experimenting with multiple neural network architectures. Additionally, all techniques produce easily-interpretable output under the form of activation maps, i.e., images of the same shape as the input image whose pixel values reflect the contribution of the input image's pixels towards the prediction of the network.

The working principle behind OM is that covering relevant sections within an image should drastically impact the classification result of the CNN, while covering background areas should influence the results less. Therefore, in order to find these relevant sections, a window is slid on top of the image with all the pixels within the window area being occluded (their values are set to 0). For every window position, the occluded image is set as input to the trained neural network and the difference in classification probability between the non-occluded and the occluded image is registered. After a complete pass throughout the image, the result is a 2-dimensional array of probability differences, where the highest differences denote the location of the relevant image sections.

A more invasive approach is proposed by GradCAM, which relies on processing the feature maps given by the last convolutional layer of a CNN. In general, after the training process, the initial layers of the CNN "learn" to recognize low-level features, such as edges, while the final layers recognize high-level features, e.g., the archaeological mound itself. Considering this, by analysing the output of the last layer before classification, the resulting feature maps should highlight the position of the most relevant features for the classification task. However, the information within the feature maps must be aggregated into one activation map that reflects the contribution of an image feature to the network's prediction. Therefore, the feature maps are weighed by their gradient with respect to the prediction result and their sum produces the final activation map.

A related technique is employed by LayerCAM, where the activation maps can similarly be extracted and weighed them by their gradients. However, as opposed to GradCAM, which performs this process only for the last convolutional layer, LayerCAM produces an intermediary activation map for every convolutional block within the network's architecture. The resulting intermediary maps are linearly combined to produce the final activation map, with higher weights assigned to the intermediary maps extracted from the later blocks of the network.

One common disadvantage that the three aforementioned techniques share is that their activation maps come at the cost of spatial resolution. Since OM aggregates

results for covering an entire area within an image and since it is not computationally feasible to slide the window every pixel, the resulting activation map is of a lower resolution than the input image. We can make similar observations both for GradCAM and LayerCAM which rely on the feature maps generated by the last convolutional layer of the CNN. Due to the image downscaling within the CNN, these feature maps have far lower spatial resolution than the input image, making the resulting activation maps also suffer from this lack of resolution.

Guided GradCAM proposes a solution for inferring high-resolution activations in the form of the individual contribution of each pixel towards the prediction. The image is first passed through the CNN, and then the resulting feature maps are passed backwards from the last layers towards the first ones. This process generates activation images containing clusters of pixels whose high values signify the presence of relevant features. However, these high-valued pixels are sparsely distributed, which makes delimiting the relevant features difficult. To ensure contiguous activations, we develop an addition to this method which we detail in Section 5.4.2.

Besides adding to the interpretability of the model's decision-making process, the activation maps can be thresholded to automatically create pseudo-annotations for more labour-intensive tasks such as the segmentation of the site area, i.e., the separation of the site from the surrounding area. The availability of cheap annotations can thus facilitate more experimentation with existing data sets and the development of more complex tools whose training would require prohibitive amounts of expert annotations. For instance, the generation of an image-level annotation for a classification task requires far less effort compared to creating a segmentation mask for the same image since the image-level label can be attributed after a relatively quick visual inspection, whereas a segmentation annotation involves the careful delineation of the site boundary. Thus, for existing classification data sets the generation of pseudo-annotations for segmentation would allow training segmentation algorithms with little intervention from domain experts.

## 5.4 Methodology

### 5.4.1 Data Preparation

For the study, we acquired CORONA KH-4B data from the CORONA Atlas & Referencing System [26]. Images from DS1105-1025 (November 1968) and DS1102-1025 (December 1967) cover the entirety of the Khabur Basin. For the initial desktop

survey, first, we orthorectified CORONA imagery and mosaicked them to generate a seamless coverage of the Khabur Basin. Second, we visually confirmed the location of 300 settlement mounds on CORONA. We also randomly picked 300 points to explore 'no-site' landscapes, and visually confirm areas that did not contain a settlement mound (Figure 5.5). Next, a custom-built script visited 'Site' and 'No-Site' locations and clipped a square chunk (1000 pixels x 1000 pixels) around each target. Image chunks were contrast stretched between 0 and 255 to exploit 8-bit data depth fully.

In the following step, we augmented data through rotation, swirling, and clipping. First, we rotated each chunk in cardinal directions to make four scenes available from

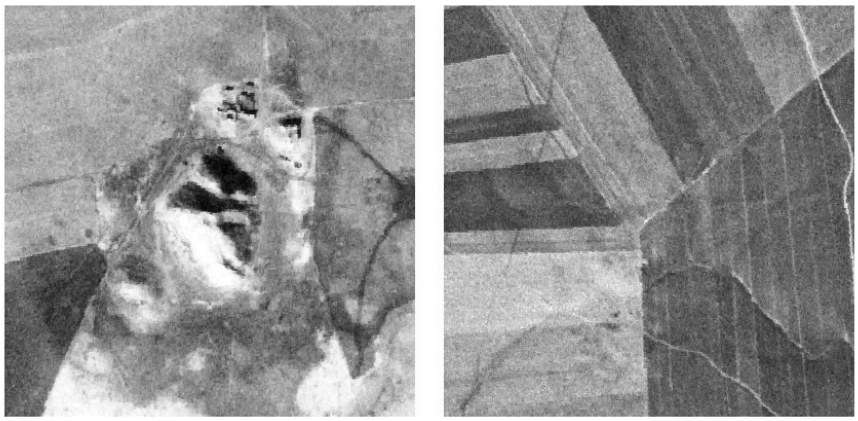


**Figure 5.5:** The binary classification scheme in this analysis. A CORONA image chunk with a site (left) and an image chunk with no indication of a mounded settlement (right).

the same area. Second, using the scikit-image Python package [152], we swirled all rotated images with a radius of 400 and with the parameter randomly determined from a uniform distribution with lower boundary of -2 and upper boundary of +2. These parameters ensure the pseudo-target generation mainly swirls the original site while keeping the background as intact as possible. With swirling, we aimed to mimic the relatively circular nature of sites; mounded settlements tend to have more circular footprints than rectangular site types. In the end, eight image chunks (four rotated and four swirled) with 1000 x 1000-pixel dimensions are further clipped into smaller pieces with 400 x 400-pixel dimensions. The clipping strategy involved "moving" the sites in four corners as well as keeping them at the centre of a scene. In doing so, the aim was to represent different parts of the immediate surroundings of the sites in additional images (Figure 5.6). This final clipping operation generated 40 images per

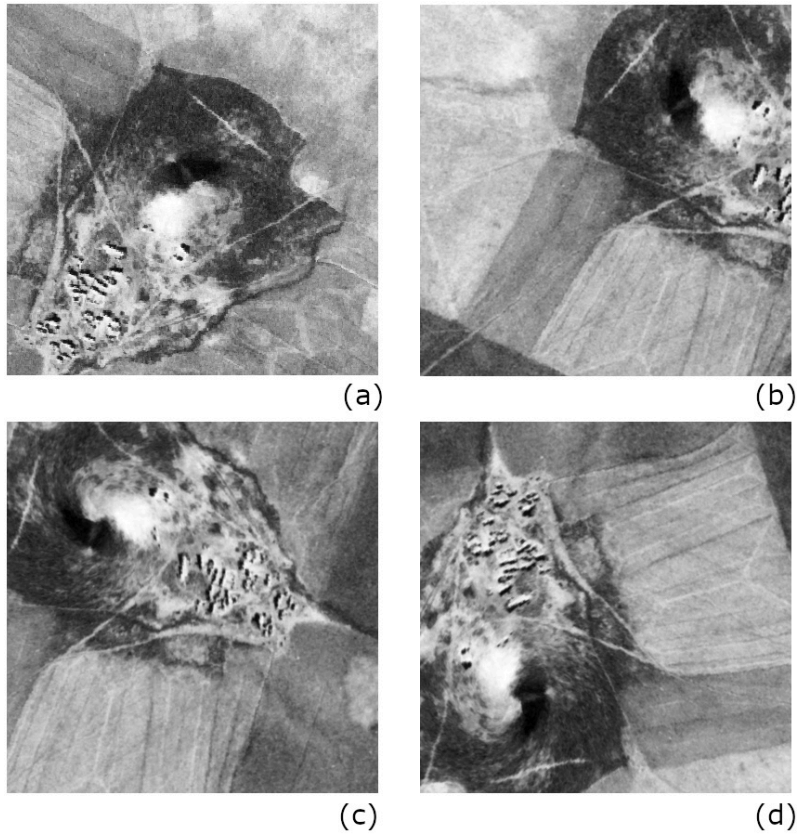



**Figure 5.6:** Four (out of 40) samples from the augmentation process are presented here. (a) the initial clip of a mound as documented on CORONA imagery, (b) clipping and rotation moves the site to the upper right corner while revealing a different background context, (c) rotated as in sub-figure b, but also moved to top left corner and swirled, (d) a different set of rotation, clipping, and swirling.

site. Therefore, we were able to gather 12,000 (40x300) image sets (binary code: 1) for 'sites'; and for 'no-sites' (binary code: 0). In total, 24,000 images were available for training.

### 5.4.2 Proposed Pipeline

For this work, summarized in Figure 5.7, we aim to utilize activation maps to derive cheap annotations that can be used for a site segmentation task as well as to formulate interpretations of the relevant areas within the images that are triggering the prediction of a network. We begin by training classification CNNs on image-level

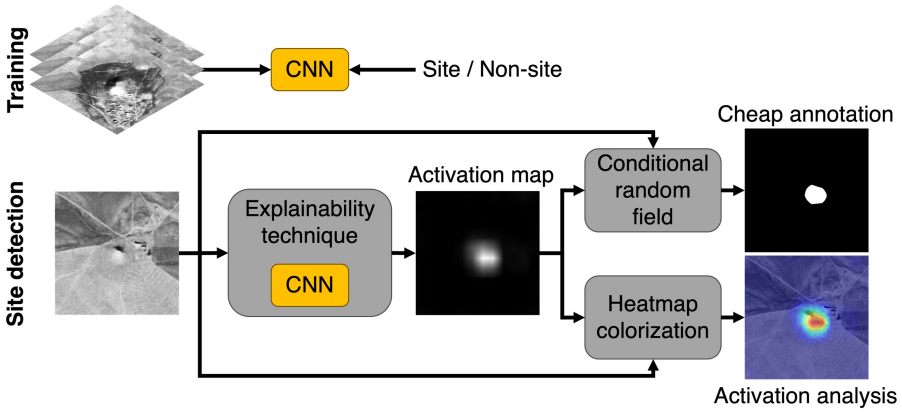



**Figure 5.7:** Workflow. We train CNNs for classifying whether a site is present or not in an image. We embed the trained CNN into explainability frameworks which we can use to both produce a segmentation mask of the site or to analyse the important image features highlighted by the network.

binary annotations, where a positive annotation signifies the presence of a site, and a negative annotation denotes the absence of the site from the input image. We utilize PyTorch [114] implementations of the three network architectures [99]. We treat the site detection as a binary classification task where the input to the network is a single-channel grayscale image and the output is a 2-valued vector, with the first value indicating the probability that no site is present in the image, while the second value indicates the opposite probability. We split our data into training and validation with an 80/20 ratio. For every type of architecture, we train 5 networks with different initialisations and a different random split of the data. During training, if we observe no improvement in the validation score for 10 consecutive epochs, we stop the process. We use the binary cross-entropy as the loss function, and we update the parameters with ADAM optimization algorithm [82].

### Generating Activation Maps

After training, we include the trained networks in the explainability tools which produce an activation map. To get more stable activation maps, we average the results from multiple initializations of the same CNN architecture. Also, for each initialization, we create a different random split between the training and validation images. We utilized Captum [85], a model interpretability library for PyTorch models to generate Occlusion Maps and to perform Guided GradCAM, whereas for GradCAM and

LayerCAM we developed our implementations based on the original papers. In all cases, the results given by the explainability methods take the form of images, where a pixel value denotes the probability that the corresponding pixel from the input image belongs to a relevant region for the classification task.

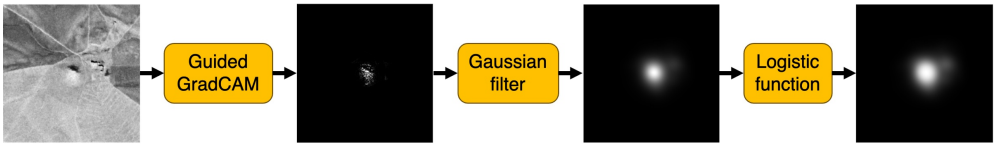



**Figure 5.8:** We propose an extension to the Guided GradCAM in order to tackle the reduced spatial resolution problem. Adding a Gaussian filter and a logistic function enhances image features that are comparable to annotations.

In addition, to address the lack of resolution of the activations generated by Occlusion Maps, GradCAM, and LayerCAM and the sparsity of Guided GradCAM activations, we propose an extension of the latter that aims to provide smoothness in activation areas, while maintaining the resolution of Guided GradCAM, which we present in Figure 5.8. We apply a Gaussian filter to smooth the pixel values of the activation image, therefore creating continuous activation areas. This, however, comes with the caveat of widening the gap in value between high-activation and low-activation areas, which can lead to a pessimistic estimation of the relevant image features. We compensate for this by passing the filtered activation image through a logistic function which creates a nonlinear rescaling of the pixel values such that previously low-activation areas would receive higher values. For visualizing the relevant features, we translate the activation map into a heatmap which we then overlap on top of the input image (see Figure 5.7). This results in a more straightforward analysis of the image features.

Here, we make observations based on visual interpretation. The aim is to build qualitative knowledge for how three different network architectures (VGG, ResNet, DenseNet) 'learn' what a settlement mound is, highlighted by four different activation techniques (Occlusion Map, GradCAM, LayerCAM, and our method based on Guided GradCAM). Our workflow includes selecting representative examples from the overall data set and exploring the activation maps on remotely sensed data.

### From Activation Map to Segmentation Mask

After obtaining the activation maps, we process them to obtain segmentation masks. In order to do this, we translate the smooth probability landscapes provided by the activation map into hard area borders by utilizing conditional random fields (CRF)

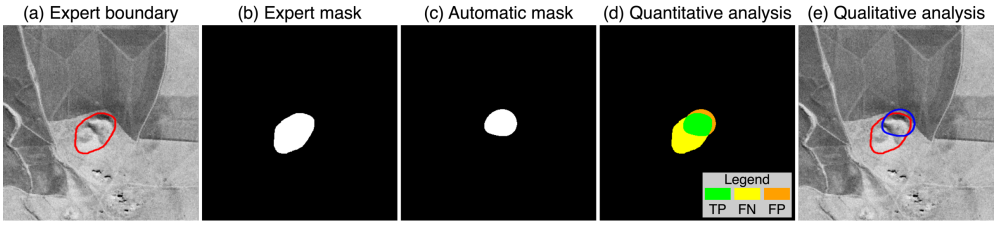



**Figure 5.9:** Example of an expert annotation (a), the segmentation mask derived from it (b), the automatically derived segmentation mask (c), the intersection of the expert and automatic masks (d), with green, yellow, and orange pixels representing true positives, false negatives, and false positives, respectively, and the intersection of expert and automatic boundaries (e).

[139]. The CRF considers both pixel probabilities from the activation map and the similarity between neighbouring pixels from the input image and outputs a binary image where the foreground corresponds to areas within the image occupied by relevant features and the background covers the rest of the image. An example of a segmentation mask generated from an activation map is shown in Figure 5.7.

We analyse the suitability of the segmentation masks produced by CRF both qualitatively and quantitatively by comparing them with site delineations provided by a domain expert. The human annotation process included drawing mound boundaries as they appear to the expert on CORONA images (Figure 5.9a). While site delineation is a subjective process, mound formation produced footprints easier to trace than many other site types and morphologies. City walls around some of these settlements also helped the annotation. To produce a quantitative analysis, we first binarize the human-annotated image such that the pixels within the boundary are assigned the value of 1, while the rest are assigned 0, creating a mask ready for further analysis (Figure 5.9b).

We then compute the Dice similarity score [45] between the binarized human annotation and the masks produced by the conditional random field (CRF) (Figure 5.9c) to assess the suitability of the automatically generated masks as annotations for segmentation. The equation describing the Dice score is presented in

$$\text{Dice} = \frac{2 \times TP}{2 \times TP + FP + FN} \tag{5.1}$$

where TP, FP, FN, refers to the number of true positive, false positive, and, respectively, false negative pixels between the binarized ground truth and the prediction (example in Figure 5.9d). For the qualitative comparison, we use the output of the

CRF to generate a boundary which we overlay, together with the human-generated one, on top of the input image (Figure 5.9e).

## 5.5 Results

### 5.5.1 Classification Performance

Since our primary aim is to explore activation maps in relation to future annotations, we only briefly mention overall model performances. We evaluated the performance by computing the precision and the recall. These metrics are computed based on the true positives (TP), false positives (FP), and false negatives (FN), calculated in the context of binary image classification, i.e., they count image-level class labels, rather than pixels. The precision reflects the proportion of relevant samples that the classification model is able to find, i.e., the proportion of correctly predicted sites among all site predictions (precision = TP / (TP + FP)). The recall, on the other hand, shows the ability of the model to find all relevant samples in the data set, i.e., the proportion of correctly predicted sites among all images with sites (recall = TP / (TP + FN)).

| Network | Precision | Recall |
|---|---|---|
| VGG16 | 0.9996 | 0.9962 |
| ResNet34 | 0.9994 | 0.9983 |
| DenseNet121 | 0.9994 | 0.9976 |

**Table 5.1:** Validation set performance of the different architectures.

In Table 5.1, we report the classification results on the validation set of the three networks. It appears that all networks learned a good fit for the data, being able to correctly classify ('site' or 'no-site') for almost all the images. All networks present similar precision, with ResNet34 showing a larger recall than the other two —albeit only very slightly. It appears that a simple augmentation technique could generate powerful classifier models with similar performances in the study area. Nevertheless, the models are trained for a very specific site type within a particular geography. Therefore, these models' generalizability is an open question; transfer learning is beyond the scope of this chapter. On the other hand, trained networks may equally perform in areas with similar relatively flat morphologies hosting settlements with mound morphologies, such as Neolithic Thessaly [7, 118].

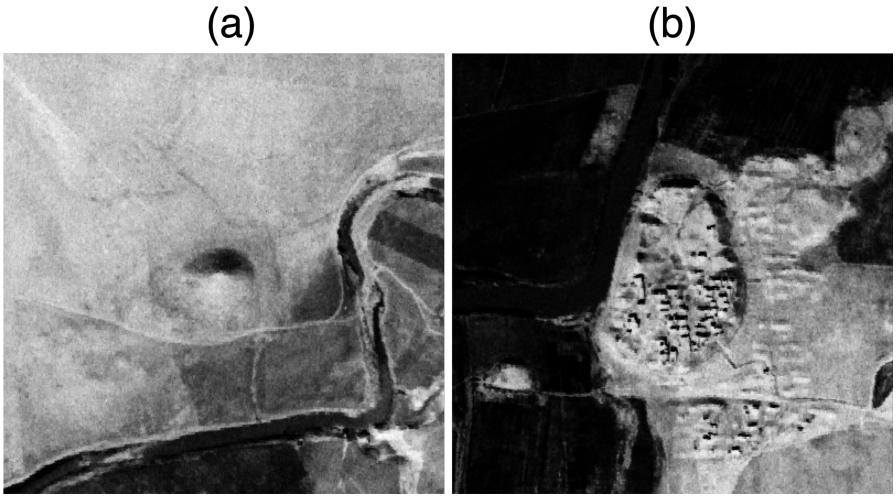



**Figure 5.10:** Two sites with simple (a) and complex (b) morphologies.

### 5.5.2 Analysis of Activation Maps

To assess the interpretability of the selected network architectures, we perform a visual analysis of the activation maps produced for sites with both simple (Figure 5.10a) and complex (Figure 5.10b) morphologies.

We begin by analysing the activations of the simple morphology (Figure 5.11). The Occlusion Map and our method fit well to the human interpretation of a site boundary for all three networks. GradCAM and LayerCAM activations exceed the site boundaries, especially for DenseNet. DenseNet produces wider activation areas, owing to its approach of aggregating information from multiple layers, thus being activated by a wider set of image features than the other two architectures. GradCAM is of particular interest since the highly activated area appears to have no immediate connections with the shape or the shadow, the two prime indications of a mound for the annotator.

Studying a more complex morphology reveals that activations can be discontinuous. In the current example, the site is dotted with modern structures, potentially adding complexity to network training. All three architectures are activated more in the north (Figure 5.12). Incidentally, this portion is cluttered less by later human occupation. It is also possible that the shadow generated more contrast against the background for the high-level features, resulting in a northerly activation. Finally, we note that only VGG is successful in identifying the smaller mound at the lower-left corner. Conforming with

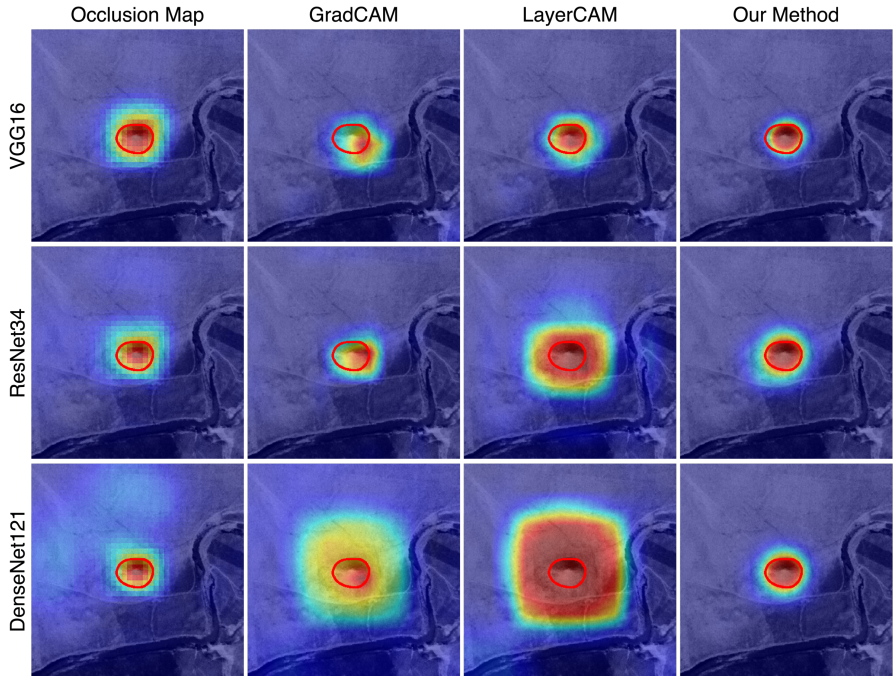



**Figure 5.11:** Activations of a single conical mound (36.832° N, 40.229° E). The red line corresponds to the expert annotation.

the previous example, the Occlusion Map and our method provide the activations that match more closely the human intuition for this smaller and circular feature.

The activations of both types of sites show that, across all network architectures, the predictions were influenced by actual archaeological features within the images. For instance, in Figure 5.11 all activations are centred on the small conical mound, whereas, in Figure 5.12, parts of the elevated area of the site are highlighted by all explainability techniques.

### 5.5.3 Activations as Sources of Annotations

For this experiment, we use the conditional random field (CRF) to process the activation maps into site segmentation masks. For ease of comparison with the expert annotations, we represent these masks as boundaries applied on top of the input image. Besides visual inspection, we also numerically assess the quality of these automatically generated masks by measuring their Dice similarity score relative to the expert annotations. We report two examples, for simple and complex morphologies. The expert

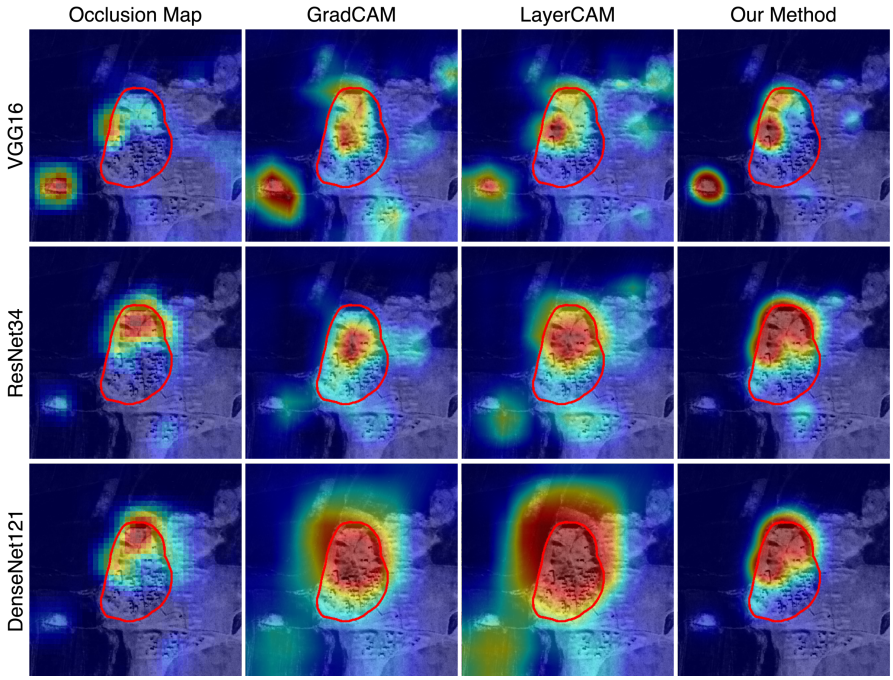



**Figure 5.12:** A feature of interest with a mound morphology (36.652° N, 40.270° E). Only the north end is clear of built environment, coinciding with most of the activations. The mound is surrounded by now dried out catchment of a branch of the Euphrates River system. This soil appears very dark on CORONA imagery. The red line corresponds to the expert annotation.

annotation (red polygon) and the results of activation mapping processed by CRF (blue polygon) are overlaid on CORONA imagery.

For a simple conical morphology, but with a more elongated extension, the network architectures variably estimate the site boundaries. We notice that the boundary generated with our method shows the highest overlap with the human annotation for all three networks, but especially for VGG and DenseNet (Figure 5.13).

The example is more telling when we study a more complex morphology and background (Figure 5.14). Adding to the complexity is how the site is represented on CORONA imagery. Image boundaries cut some parts of the site as it does not fit into the predetermined image chunk. Jakoby [72] discusses if Tell Mosti with a 'cup-and-saucer' shape exhibits morphological characteristics of a Kranzhügel type [135]. To bypass the site representation problem, the human annotation only considered the 'cup' as the 'site'. Once again, our method is able to determine the extents of the

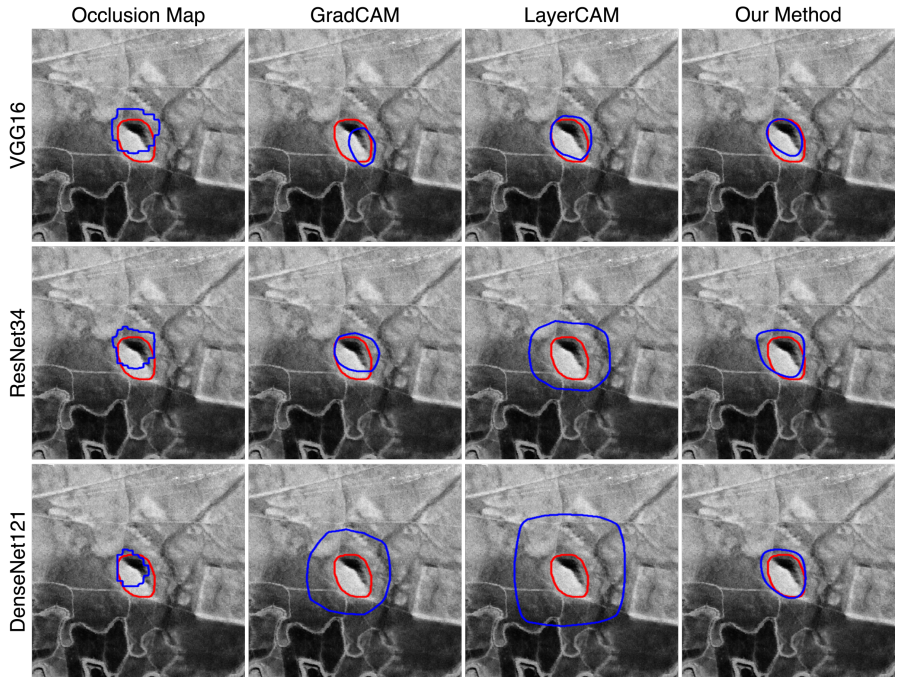



**Figure 5.13:** A conical mound (36.639° N, 40.977° E) and its model predictions. Note how they all miss the rectangular site just next to the mounded settlement. The red line corresponds to the expert annotation, while the blue line is the predicted boundary.

site, but only for DenseNet and ResNet. These networks are clearly archaeology agnostic, but still conforming with the visible boundaries of the 'cup'. We discuss the image-cutting site boundaries in the next section as we evaluate biases in the training dataset.

| Network | Occlusion Maps | GradCAM | LayerCAM | Our Method |
|---|---|---|---|---|
| VGG16 | 0.4483 | 0.4083 | **0.6043** | 0.5928 |
| ResNet34 | 0.4826 | 0.5149 | 0.4041 | **0.5954** |
| DenseNet121 | 0.4514 | 0.3537 | 0.213 | **0.6185** |
| Mean Values | 0.46 | 0.43 | 0.41 | **0.60** |
| Variances | 0.0004 | 0.0067 | 0.0383 | **0.0002** |

**Table 5.2:** Dice similarity between the predicted and annotated site area over the entire data set. The bold values represent the highest score achieved per network.

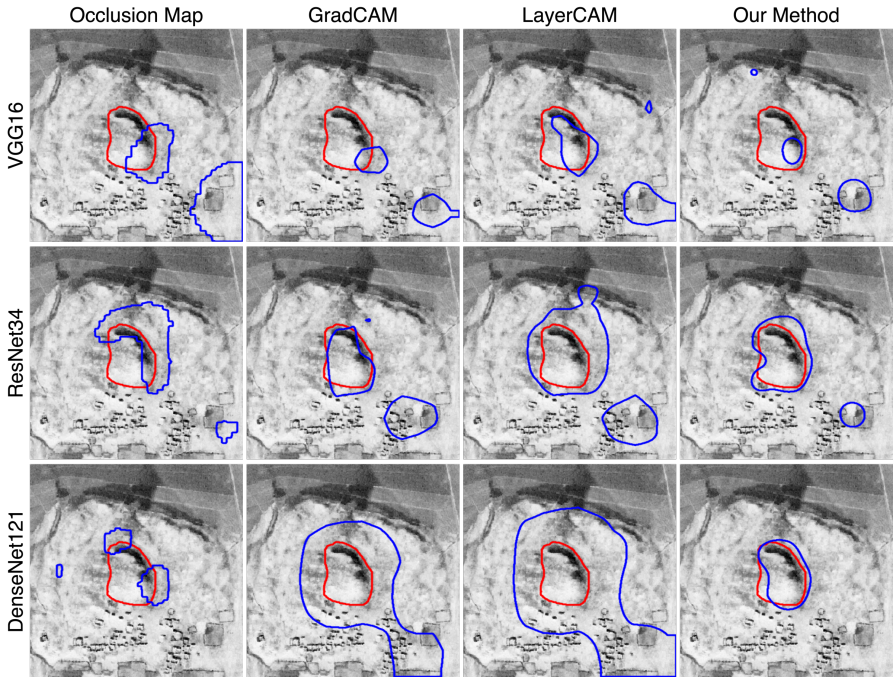



**Figure 5.14:** Tell Mosti (36.624° N, 41.615° E) exhibits a more complex case. Please note the actual site is larger than the digitized/annotated "crown". The image extent cuts the site due to its size. The red line corresponds to the expert annotation, while the blue line is the predicted boundary.

Finally, we provide a quantitative analysis to show an overview of the quality of the generated masks over the entire data set. We report the Dice similarity scores between the predicted boundary and the annotation in Table 5.2. We observe that although all three networks perform similarly for the classification task, their ability to delineate the boundary of an archaeological site differs. The variations in performance possibly stem from architectural differences between networks, as well as from the type of explainability method we employed. One notable exception is given by our method, which shows the least amount of variation between Dice scores across networks. Also, the masks generated from processing the activations of our method produce the highest individual score for DenseNet and better average performance across all architectures than the masks generated from the other explainability techniques. Thus, our extension to the Guided GradCAM appears to be a robust annotation generator for the given training data set collected from this particular geography.

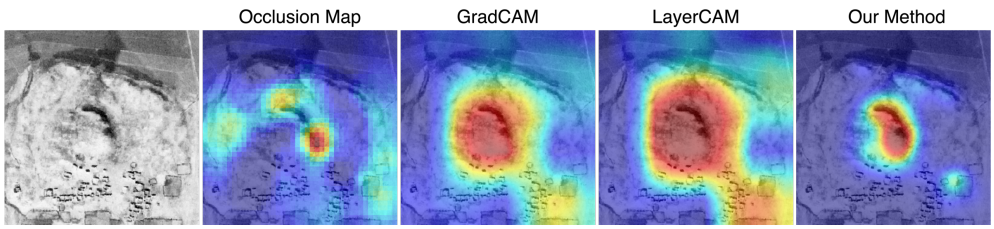



**Figure 5.15:** Different activation maps for DenseNet. Please note how our extended Guided GradCAM method identifies some parts of the cup of the site matching the human annotation. The slightly activated area in the right left is much more pronounced in GradCAM and LayerCAM. The red line corresponds to the expert annotation.

## 5.6 Discussion

Here, we present the activation maps of multiple sites with varying morphologies, and we discuss the potential of the activations as sources for cheap annotation. We also derive interpretations of these activations to understand how different CNN architectures learned to distinguish archaeological sites.

We start our discussion using the previous example from Tell Mosti. As we discussed above, the human annotation only included the 'cup' of the site, so there is a clear mismatch between human annotation and model estimation for the most part. It is only that our proposed addition to the Guided GradCAM estimates an area close to human interpretation, but GradCAM and LayerCAM reveal a high-activation area in the lower-right corner of the image (Figure 5.15). To investigate, we explored a high-resolution digital elevation model of Tell Mosti (Figure 5.16). Overall, higher elevations roughly overlap with the results of activation maps. In this particular case, we observe the benefit of analysing the activation maps since they indicated the south-eastern extension, which the expert missed since it is not immediately visible on the CORONA imagery.

To showcase how activation maps may relate to non-circular site morphologies, we selected Tell Jamilo (Figure 5.17a) and Tell Hadi (Figure 5.17b) which present comparable morphologies. The orientations differ but their tangled morphologies are similar. The activation maps of both sites show that VGG predictions are more strongly triggered by round features, a characteristic that many mounds within our data set share. We observe a similar pattern for VGG in the activation map of Figure 5.12. This reliance on round features can be due to the loss of context information after each pooling operation is performed from one group of layers to the next. On the other side, ResNet and DenseNet still retain context information even in the deeper layers

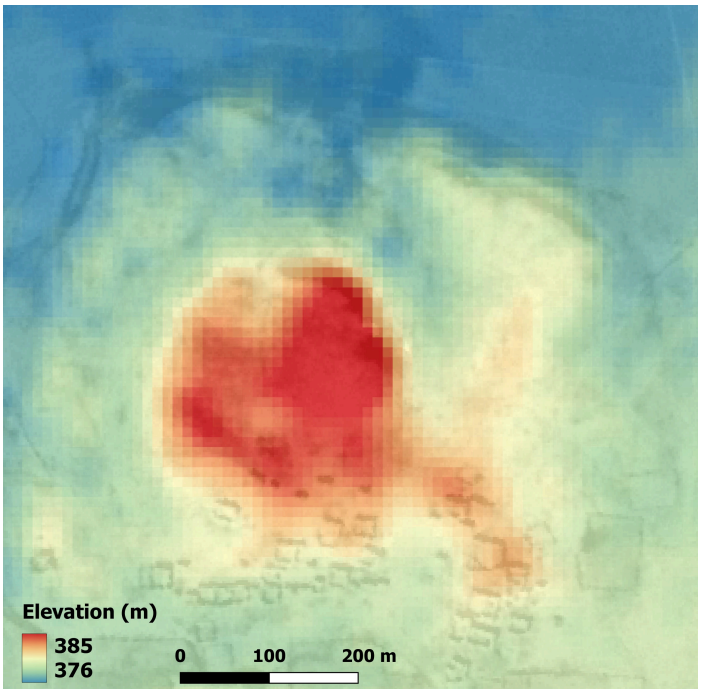



**Figure 5.16:** The digital elevation model of Tell Mosti highlights the elevated core of the site. Manual annotation considers only the highest eastern blob but misses the entire core.

by using skip and, respectively, dense connections in their architectures, making their activations a promising source for generating segmentation masks.

Furthermore, we explore single conical and complex morphologies in the same image chunk. Figure 5.18 contains two sites in the same image frame, the larger more complex one being identified as Tell al-Shur [128]. Because the sites are close to each other, small CORONA image chunks covered both. We initially identified them as different sites, so the script created one image case for each site with greatly overlapping backgrounds.

In the first instance (Figure 5.18a), the larger site with more complex morphology is at the centre and the smaller conical site is slightly to the right. We observe that DenseNet is able to highlight both sites with GradCAM and LayerCAM, matching the human interpretation. Also, our method is particularly convincing as an annotation source since it activates both sites at the same time with relatively good coverage of the archaeological features without including much of the surrounding landscape. The same couple produces different activations when the central focus is shifted (Fig-

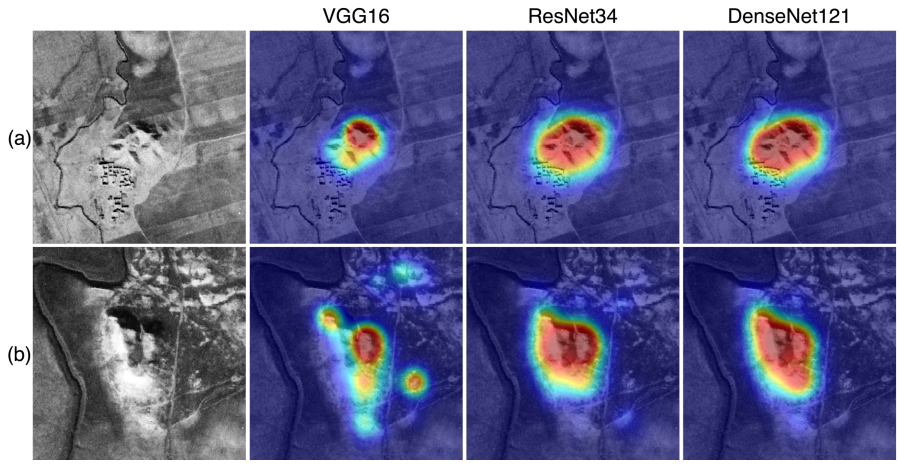



**Figure 5.17:** Activations generated with our method for Tell Jamilo (36.683° N,40.607° E) (a) and Tell Hadi. (36.870° N,41.865° E) (b).

ure 5.18b). As in the previous case, the circular site is represented almost in entirety. However, it appears as Tell al-Shur lost significance in the activations. It is possible that all three networks learned that a mound should be circular, and our swirling augmentation further emphasized circularity. It may be also possible that models were influenced from the location of Tell al-Shur within the image. In this case, the site is located at the edge of the image, suggesting that the contribution of a site to the prediction in a multi-site image is dependent on the site's position within that image. This is expected since the networks are trained for classification, which does not incentivize the activation of all archaeological features present in an image, but rather of the strongest visual cue, which, in this case, is the circular small mound. When both sites are fully included in the image but shifted upwards from the centre (Figure 5.18c), we notice similar activation patterns as in Figure 5.18a. This mainly shows the invariance to image shifts, a general characteristic of CNNs due to their usage of pooling layers. The difference from the activations in Figure 18b shows that this invariance still requires the relevant image features to be entirely present in the image.

Throughout our analysis of activation maps, we noticed that LayerCAM, through its aggregation of activations from multiple layers, produces the widest coverage of the archaeological features, which proves especially useful when multiple sites are present in an image. GradCAM, by focusing only on the final layer, trims this wider context which results in more focused activations, but at the cost of ignoring some

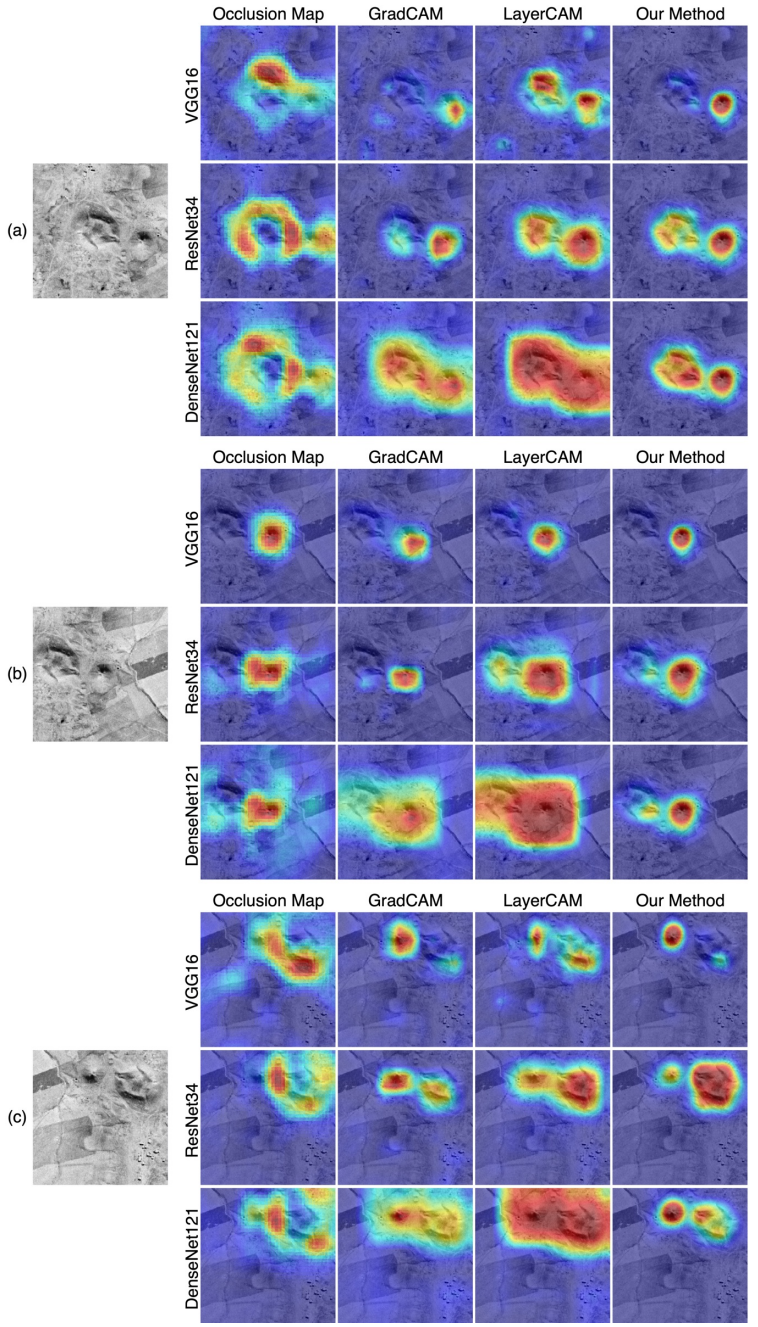



**Figure 5.18:** Different activations of Tell al-Shur (36.845° N, 40.458° E) with a complex morphology and a nearby conical site. Different augmentations of the same region are activated in different ways.

archaeological features, as is the case for ResNet in Figure 5.18. The Occlusion Maps mainly highlight differences in elevation, by signalling shaded areas as discriminatory features. This technique, however, suffers from a high computational cost, as it requires a new network prediction for every position of the occlusion window, its output is of low resolution, and it also requires choosing a window size, since small occlusion windows may not significantly change the prediction score. Our method generally produces activation maps that include even less of the surrounding landscape than GradCAM which enable a clearer observation of the relevant archaeological features within the images. Moreover, given their sharp boundaries, the activation maps generated with our method can become a strong basis for producing annotations for segmentation as also indicated by the results in Table 5.2.

When it comes to the network architectures, we notice that VGG seems to rely less on context, i.e., image features characterizing the whole site, focusing more on general features, such as roundness. On the other hand, ResNet and DenseNet appear to base their predictions on increasingly more contextual information due to their connections that forward information from previous layers to the following ones, while VGG lacks this characteristic. This wider coverage of the archaeological site by ResNet and DenseNet activations can mean that these two architectures may show better adaptability than VGG to changes in site morphology when, for instance, the geographical area changes.

## 5.7 Conclusion

Exploring how different networks are activated for mounded settlements proved to be a fruitful exercise. The study generated voluminous data, and we followed a particular path in interpreting experiment results. Therefore, the topic is open, and many other inferences can be made. Our aim has not been to develop a "best-practice guide" with detailed accuracy statistics and thresholds. Inferences we made in this chapter depended upon our CORONA-specific training dataset with a specific site morphology.

The results we reported here are not benchmarks for any network or an activation method. The settlement mound has a particular morphology uniquely contextualized in Upper Mesopotamia. Therefore, our interpretations are specific to the training dataset, and we try to avoid making broad statements. However, experimenting with network architectures using different activation techniques appears to be a fruitful exercise and the workflow may be generalizable.

Our work, while only emphasizing coarse associations between settlement mor-

phology and periodization, opens the door to more detailed and systematic analyses through the application of deep learning. The widespread presence of mounds suggests an opportunity to extend computer-assisted morphological analysis, with our study serving as a step in that direction. Additionally, our approach finds utility in a detection mechanism, where users can observe highlighted regions as potential archaeological sites within large geographical areas.

Furthermore, we showcase the potential of using activation maps as the basis for producing cheap annotations, which, with the incorporation of corrections, either through user intervention in an active learning setup [123] or automatic adjustments [159], can contribute to refining predictions and improving the overall accuracy of algorithms for site delineations. For this particular region, we observed that DenseNet in conjunction with our modified version of Guided GradCAM produces the most accurate site annotations. Moreover, DenseNet's usage of wide contextual information may indicate good robustness to potential changes in the site's morphology and in its surrounding landscape.

Finally, despite our initial focus on settlement mounds and exclusion of periodization concerns in our preliminary experiments, our method holds promise as a potential deep learning-based expert helper system for assisting desktop surveys. Additionally, the integration of Digital Elevation Models (DEM) into the workflow could amplify our method's potential for morphological classification, presenting a versatile tool for archaeological analyses. Also, to better assess the applicability of this study, we aim to expand it by including a more diverse set of site morphologies.