Architectural Distant Reading

Using Machine Learning to Identify Typological Traits Across Multiple Buildings

Cecilia Ferrando 1, Niccolò Dalmasso 2, Jiawei Mai 1, Daniel Cardoso Llach 1

1 Carnegie Mellon University, Computational Design Laboratory
cferrand@alumni.cmu.edu
jiawei@andrew.cmu.edu
dcardoso@andrew.cmu.edu

2 Carnegie Mellon University, Department of Statistics & Data Science
ndalmass@stat.cmu.edu

Abstract. This paper introduces an approach to architectural “distant reading”: the use of computational methods to analyze architectural data in order to derive spatial insights from—and explore new questions concerning—large collections of architectural work. Through a case study comprising a dataset of religious buildings, we show how we may use machine learning techniques to identify typological and functional traits from building plans. We find that spatial structure, rather than local features, is particularly effective in supporting this type of analysis. Further, we speculate on the potential of this computational method to enrich architectural design, research, and criticism by, for example, enabling new ways of thinking about architectural concepts such as typology in ways that reflect gradual variations, rather than sharp distinctions.

Keywords: Architectural Analytics, Machine Learning, Classification, Religious buildings, Space Syntax

1 Introduction

An algorithmic analysis of architectural data can yield valuable insights about spatial configurations and typological traits. In this paper we use machine learning techniques to study spatial configurations across multiple examples in an architectural dataset comprising floorplans of religious buildings including mosques and monasteries. The dataset codifies these buildings’ spatial structure, areas, connectivity, among other characteristics, as a vector of features. Analyzing this dataset using machine learning techniques we are able to successfully classify the buildings as either mosques or monasteries, and derive other architectural insights concerning the links between typology and spatial connectivity. Graph-based comparisons at the building level were
the most useful in our classification, highlighting the importance of spatial structure in defining typology.

Recent work in the fields of cultural analytics and digital humanities has employed computational methods to analyze large “cultural” datasets including, for example, literary, visual, and social media data [1-3]. Instead of focusing on close readings of an individual text or work of art—as the literary and art historical traditions dictate—work in these emerging fields mobilizes computational methods to deal with larger bodies of work (e.g. an entire literary genre, or the collected works by a single artist or author). This “distant reading” [4-5] approach enables the discovery of patterns of stylistic change, or the gradual emergence of linguistic or visual traits, across large bodies of work. In architecture, a similar approach has been used to study the building design process based on multiple data collected during design coordination [6]. In this paper we introduce a novel computational method to analyze a dataset of buildings in order to study typological traits and derive insights about their spatial configuration: an architectural “distant reading”.

The paper discusses the computational methods employed, and our findings. It further suggests some implications of this research for broader architectural analysis—for example as a tool for studying typological similarity and variation across large bodies of architectural data, and as a method to search, generate, and ground architectural hypotheses—suggesting a path towards a computationally enriched practice of architectural design and criticism.

2 Methods

2.1 Spatial analysis

Important to our approach is a tradition of spatial studies that mobilizes techniques from mathematics and computation to study architectural and urban configurations [7-13]. In particular, two spatial analysis instruments play a salient role in our study: graphs and isovists. Graphs represent spatial hierarchies in the form of nodes (rooms) and edges (connections), which can then be treated mathematically. Architect and mathematician Christopher Alexander, for example, studied the applications of graph theory to the analysis of urban connectivity [7] and function [8]. Isovists, on the other hand, are measures of the visual connectivity of a particular space [11], and thus offer an important qualitative insight about architecture. Examples of the use of the use of isovists for architectural analysis can be found in [12] and [13]. A first attempt at combining these spatial analysis techniques with machine learning methods can be found in [14].

2.2 Data collection, curation, and pre-processing

Our dataset comprises raster images of plans of monasteries, abbeys, and mosques collected from Wikimedia, Google search, and .dwg files from the architectural database ArchNet [15]. Both mosques and monasteries are sampled from a variety of geographic and historic situations. We curated the dataset by discarding trivially simple plans (defined as plans comprising only one or two rooms), which we
considered outliers that would likely skew the analyses. Samples of the raw data are included in Figure 1. In order to create a consistent dataset including areas, spatial connections, geometry, and isovists for each building, we defined and followed a two-step protocol, available in [16]. The first step was to use 3D modeling software Rhinoceros to segment building plans into distinct spaces. The second step was to extract isovists and connectivity information from each space using a custom Python-Grasshopper workflow we developed [16], which streamlined the process significantly. The protocol included guidelines determining, for example, the level of detail needed in the drawings, and ways to resolve certain spatial ambiguities. Examples of processed plans are reported in Figure 2.

Fig. 1. On the left: the plan of Köse Hürev Paşa Camii (mosque) in Van, Turkey. The plan is a .dwg file from the ArchNet mosques dataset. On the right: the plan of Morimondo Abbey, Morimondo (Milan), Italy. The plan is a raster image from Google Images.

Fig. 2. Top, from left to right: the plan of Köse Hürev Paşa Camii (mosque) in Van, Turkey, its segmentation and graph representation. Bottom, from left to right: the plan of Morimondo Abbey, Morimondo (Milan), Italy, its segmentation and graph representation.
Following a “supervised learning” or “classification” approach, our dataset includes information about each building such as area, connectivity, isovists, and whether the building is a mosque or a monastery. As shown in Figure 2, our building graphs comprise a series of nodes (rooms) each indexing the area, and isovist area, plus the overall structure to which nodes are connected, as attributes. The final dataset comprises 39 buildings and 1,600 rooms, and is publicly accessible in [16].

2.3 Exploratory data analysis

This section includes an analysis of the features we extract and visualize from the dataset, followed by an in-depth description of the classification performances for the algorithms in both room and building-level analysis.

![Fig. 3. Matrix plots of the three features (number of connections, area and isovist) and building type for all the rooms available in the dataset (around 1,600), with monasteries and mosques in red and blue respectively. Densities are shown in the main diagonal, scatterplots in the lower triangular part, correlation in the right triangular part and boxplot distributions in the right furthermost column.](image-url)
Exploring the variables at the room (node) level—area, isovist, and number of connections—offers a local perspective of the spatial structure. Figure 3 illustrates these three features, colored by building type: red for monasteries and blue for mosques. In all the plots, one single point represents a node of the graph, of a total of around 1,600. For visualization purposes, we have taken the logarithm of each feature. On the main diagonal, we see the density plot for each of the features by building type, with the bottom right position being just a histogram of the available points for each category. The lower triangular part of the plot provides the scatterplots of the variables, while the upper triangular part reports correlation between features both across all the available data and between only monasteries and only mosques. Finally, the furthest column provides a density comparison in terms of boxplots.

The analysis shows a slightly positive correlation between area and number of connections and between centered isovist and area for both monasteries and mosques. These two observations are in line with what observed in the architectural plans, as one might expect the number of connections and the isovist for rooms to grow the larger the room gets. In order to test whether the distribution of the features does not change between monasteries and mosques we employ the Kolmogorov-Smirnov test (KS test), which is a non-parametric statistical test—i.e. makes no modeling assumptions on the underlying distribution. We set our confidence level at 95%, which corresponds to a p-value of 0.05.

![Feature Space - Room Level Classification](image)

**Fig. 4.** 3D scatterplot of the monasteries and mosques, in red and blue respectively. On the three axes, the number of connections, area and centered isovist area.

This analysis shows that the rooms with the highest number of connections to other rooms are more likely to belong to mosques. Additionally, rooms in monasteries seem to be slightly larger in size. The isovist data suggests that the distribution of monasteries is slightly bimodal, but without significant difference as highlighted by the KS test. Figure 4 shows a scatterplot of the three features in 3D space defined by the number of connections, area, and isovist. Keeping in mind that our experiment involves a machine learning algorithm tasked to separate the different-colored points in this space, we can see how this task seems to be highly non-linear, varying in
different regions of the space, and hence difficult and unlikely to be tackled successfully by the human eye only.

2.4 Algorithmic methods for classification and graph comparison

We conducted a series of preliminary analyses training different machine learning algorithms on our dataset with the goal of determining their accuracy in classifying each building either as a mosque or as a monastery based on the information available at the node (room) level, leaving aside the graph codifying the building’s spatial structure. Specifically, we used the following algorithms: a) nearest neighbors, which looks for the most prevalent type of points in the vicinity of the feature space, b) logistic regression, which fits a linear model for the logarithm of the odds of a point to belong to a specific building type, c) gradient boosted trees in the XGboost implementation, and d) random forest. The last two algorithms, which rely on an ensemble of decision trees, yielded the best results, as reported in Table 1.

While the room approach outlined above yielded a better-than-random accuracy in the classification, our key approach, however, was to develop a new method to algorithmically analyze the global spatial structure of the buildings in the dataset, codified as graphs, in order to perform a comparative building-level analysis. This approach was extremely successful in classifying building types, and is the focus of the following sections.

We used a graph-based kernel built in [17] to quantify the similarity between building graphs while preserving graph structures. Based on the intuition that random walks on two similar graphs (plans) should be similar as well, this kernel determines the degree of similarity between two graphs by looking at the behavior of random walks of the same length over two graphs. Here performing a random walk means randomly starting on a node in the graph, move with equal probability in any of the connection of the node, and repeating this process with a probability $p$ of stopping the random walk at any given time. As there is a closed-form solution to compare random walks of any length between two graphs, we use such similarity score between random walks as the kernel value, storing them into a symmetric matrix $K$ of size $g \times g$, where $g$ is the number of available graphs, formally known as the “Gram matrix”. We leverage this matrix in our use of Kernel SVM [18,22]. SVM builds a decision boundary in the feature space such that the distance between such boundary and points from either of the two labels is maximized. With the use of a kernel, such decision boundary is placed in a higher dimensional space, in which one point is in our case a graph. By looking at the graphs which are either mis-classified or the closest to such boundary, we can get an understanding of which spatial situations are the most useful for classification.
Table 1. Accuracy results for room-level analysis, comparing all algorithms to the random classifier, which identifies a room as belonging to a mosque regardless of the input features.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Test Set Accuracy (%)</th>
<th>Performance vs. Random Classifier (Relative %)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random Classifier</td>
<td>53.66</td>
<td>-</td>
</tr>
<tr>
<td>Nearest Neighbours</td>
<td>67.57</td>
<td>+10.1</td>
</tr>
<tr>
<td>Logistic Regression</td>
<td>55.6</td>
<td>+3.6</td>
</tr>
<tr>
<td>Logistic Regression - Lasso</td>
<td>55.6</td>
<td>+3.6</td>
</tr>
<tr>
<td>Random Forest</td>
<td>72.01</td>
<td>+34.2</td>
</tr>
<tr>
<td>Gradient Boosted Trees</td>
<td>69.11</td>
<td>+28.8</td>
</tr>
</tbody>
</table>

2.5 Evaluation

Evaluating the performance of our building-level analyses posed challenges given the limited size of the dataset. Rather than splitting the dataset into a “train set” and a “test set”—an assessment method well suited for tasks with large sample sizes [19]—we employed the “leave one out cross validation” (LOOCV, [20]) method, which has been shown to closely approximate the error of the split dataset approach [21]. We measure the accuracy of our algorithms against the trivial random guess based on the proportions of the classes. For every group of parameters, we split the training sets into \( k \) groups, routinely holding out one of them, training on the other \( k-1 \) groups and calculating the accuracy on the held out group. This does not allow us to search for the best parameters set (“hyperparameter optimization”) for the different algorithms we use, using the implementation default.

3 Results

Intuitively, the spatial structure of a building is one of its key features, and perhaps essential for accurate classification. Using a “kernel trick” with a graph kernel based on random walks, as described above and introduced in [17], we were able to preserve the graphs and thus compare buildings based on their whole spatial structure, avoiding information loss. To the best of our knowledge, it is the first time that such approach is used on architectural data.
Remarkably, using this method the LOOCV classification accuracy is 93.74%, which is an improvement of 87.5% with respect to a random classifier. It is also a great improvement from the room-level analysis which reaches a 72% accuracy. This analysis focuses exclusively on the graph, which codifies the building’s spatial structure. Area or isovist information is not included. For computational reasons we do not use the largest mosque in the training data (which has more than 130 nodes), leaving 19 monasteries and 19 mosques for classification. The graphs corresponding to the plans in our dataset are reported in Figures 5 and 6. The successful classification confirms that a building’s spatial structure codified as a graph has great analytical potential for architectural “distant readings”.

One of the most interesting results of this analysis is the discovery of what we may call typological gradients. For example, Figure 6 shows the graphs of the two mosques which were mis-identified as monasteries. The mosque on the right-hand side has an extremely simple structure, while the one on the left-hand side includes signature spatial structures found in both other monasteries and mosques (respectively, enfilades and “star-like” branching), hence the reason for misclassification. This is also highlighted by the fact that such mosque is a support vector of the SVM, shown among the closest points to the classification boundary (Figure 7). The graph of the other misclassified mosque presents both a “flower-like” structure and a more connected section. While the first characteristic is present in both building types, monasteries seem to have a more complicated and connected structure, which could represent a potential issue in classification. Reinforcing the first observation, the bottom left mosque closest to the boundary in Figure 7 also presents a “flower-like” structure, as it is present in both building types. Overall the closest graphs to the decision boundary are quite simple, suggesting that more complicated patterns are further separated in the graph space and, more importantly, more descriptive and distinct of the building type.

It is also interesting to observe the farthest graphs from the decision boundary – the ones that the algorithm classified more easily (Figure 8). The two farthest monasteries share the spatial characteristic of having long enfilades of rooms with limited branching (and a singleton terminal node); in general, they seem to have a more “in-depth” spatial structure. In the two farthest mosque plans, we notice the peculiar existence of rooms with a very large number of child-nodes: in such cases the child-nodes tend to also be terminal nodes.
Fig. 5. Graph representation of the architectural plans of all the 36 correctly classified graphs.
Fig. 6. Graphs of the mis-classified buildings by Kernel SVM, both of which are mosques.

Fig. 7. Closest graphs to the decision boundary of the Kernel SVM, five mosques and one monastery. The top two left graphs are the support vectors.
Fig. 8. Farthest graphs from the decision boundary of the Kernel SVM, five mosques and three monasteries. From top-left to bottom-right in order of distance from the decision boundary.

3.1 Limitations

Computational methods—especially those labeled as “machine learning”—are often accompanied by a rhetoric that emphasizes the automated aspects of the work, while hiding the laborious processes that make it possible. To avoid this common pitfall, we want to highlight for readers how laborious and time intensive the process of collecting, aggregating, curating, and processing the data for these analyses was. It involved collecting a suitable set of examples from different sources, and retracing many raster images into vectorial representations, which could be processed following the steps detailed in [16]. This was not simply a mechanical process: plan preprocessing can only be successfully executed by trained architects, who are able to understand the visual codes and interpret the ambiguous situations found in many of the plans. Preparing the data for this project thus relied on long hours of drafting, labeling, and revising work by a team of committed research assistants.

And yet, the small size of our dataset makes any generalization difficult. While the paper proposes a novel and sound method and workflow for architectural “distant readings”, it is best to treat its specific findings (e.g. about the architectural differences between mosques and monasteries) as informed architectural hypothesis rather than as scientific “truths”. These computationally informed hypotheses, however, may be corroborated or refined by expanding the dataset. Here, computational complexity is also currently a limitation. In calculating the graph kernel, a very large matrix needs to be inverted, involving a complexity of the order of $O(n^5)$. In other words, calculating the value of the kernel for two graphs with 100 nodes, the methods invert a matrix with 100 million elements, which can be computationally expensive. To account for that, we limit the calculation of the kernel to graphs with less than 100 nodes. Considering the fact that the number of kernel computations scales quadratically with the number of graphs in the training set, this method may not be practicable with much larger datasets. This limitation might be tackled to an extent by obtaining access to a supercomputer.

4 Conclusion

This paper has outlined an approach to architectural “distant reading”: the use of computational methods to analyze architectural data in order to derive spatial insights from—and explore new questions concerning—large collections of architectural work. The evidence assembled above shows that buildings’ spatial structures codified as graphs have a great analytical potential which can be used to automatically distinguish different types of buildings, as well as to explore other architectural characteristics. Remarkably, our method to classify building types based on their spatial structures performed at 93% accuracy. While our dataset focuses on religious complexes, we are confident that our approach can be transferred to other building types, and be used to address other architectural questions.

The success of our method for classifying building types raises one important question: what combination of spatial features makes the classification so accurate despite the high “in-culture” variability that characterizes our dataset? In other words: do monasteries and mosques have an underlying spatial “fingerprint”? Figures 6-9 offer some materials to start answering these questions. The spatial characteristics that
appear to have contributed the most to the definition of a decision boundary are, in monasteries, the existence of “in-depth” spatial structures with enfilades of single nodes; in mosques, the nodes with a large number of terminal children in a “star-like” structure (Figures 8-9). The analysis of our data thus suggest that monasteries tend to privilege spatial depth, a spatial mechanism that creates sequences of progressively more segregated spaces; mosques tend instead to privilege high-branching nodes with a large number of child-nodes, which reflects the existence on these plans of series of spaces sharing the same hierarchical status in terms of accessibility. In short, compared to monasteries' vertical spatial hierarchy, mosques feature a more horizontal one. These spatial traits correspond to distinct kinds of architectural connectivity (and experience), which an expert may be able to observe in the drawings (Figure 9). This paper thus offers an innovative method for the quantitative assessment of this spatial and typological trait. On a technical level, the application of Kernel SVM to architectural plan data, which enabled us to use preserve graph data in our algorithmic analysis, is—to the best of our knowledge—a novel contribution of this paper.

Rather than a replacement for qualitative observation, we propose these methods as ways to expand the analytical repertoire of architectural designers, scholars, and researchers. In combination with other forms of analysis, architectural “distant readings” can help make visible spatial traits across large collections of data, explore new questions concerning architectural concepts such as typology, and offer quantifiable metrics which may help ground architectural hypotheses. As architectural scholars, for example, we may find new insights by comparing variations in large datasets. Changing conceptions of architectural intimacy, to give but one example, may be traced with increased eloquence across cultures, continents, and/or timeframes. As architectural designers, we may be assisted by the new kinds of context offered by these methods. We may, for example, study how a specific building design compares to a host of architectural precedents, or assess its “fit” within a specific architectural culture or typology. We may also navigate architectural data in new ways, finding unexpected spatial connections in buildings belonging to different traditions, or serving different functions. The very concept of typology can thus be enriched, and nuanced.

References

15. ArchNet .dwg collection, Drawings of Islamic Monuments, archnet.org/collections/843
16. github.com/c0deLab/ML-architectural-analytics