Hierarchical Visual Feature Analysis for City Street View Datasets

Lezhi Li, James Tompkin, Panagiotis Michalatos, and Hanspeter Pfister

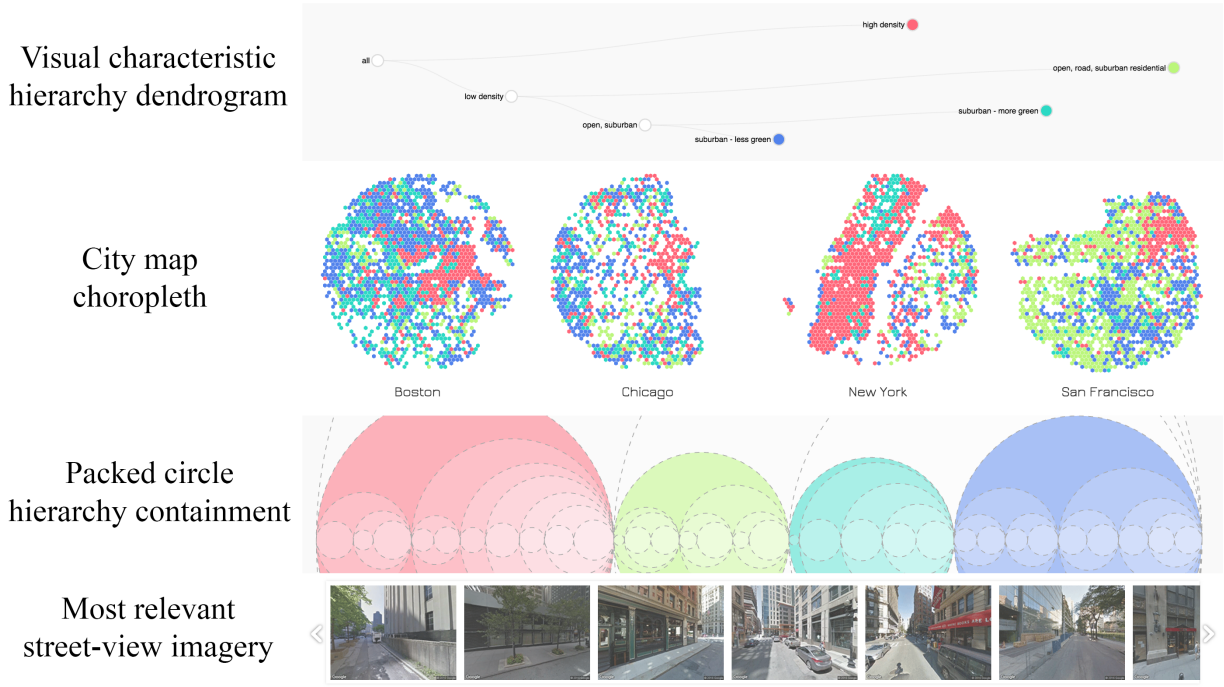

Fig. 1. Exploring 'perceptual neighborhood' with our hierarchical clustering of visual features for geographically-embedded images. From top to bottom: tree representation of the hierarchy as an interactive dendrogram; geo-location of the images as cloropleth map; circle containment representation of the hierarchy; street-level imagery for geographic regions of the selected hierarchy levels.

Abstract—The visual appearance of city neighborhoods helps us to mentally map urban spaces. For instance, from the visual features of a city or neighborhood, we gain perspectives on local identity as might be described by their functions, demographics, or affluence. An effective way to summarize and present this information would be useful, e.g., for urban design and planning. We explore whether these perspectives can be automatically learned from street-level imagery using a deep neural network and build a visual analytics tools to explore what is learned. Starting with a dense geo-sampling of city Google Street View data, we train a neural network to learn visual features. Then, we cluster these features using unsupervised learning to build a similarity hierarchy of visual appearance. Existing approaches for exploring this kind of geographically-embedded cluster data often have difficulty in addressing the need to compare across both the visual hierarchy and the geography of the different neighborhoods. To improve this situation, we develop a visualization scheme which allows users to keep track of both the geographical and semantic interpretations of the data. In doing so, we aim to provide an exploration tool to aid in the visual study of urban environments.

Index Terms—Machine learning, Hierarchy data, Visualizing spatial and non-spatial data, Coordinated and multiple views.

1 Introduction

As a first-time visitor to a city, walking the streets and seeing the visual characteristics of the neighborhoods provides rich insights into local functions, identity, demographics, affluence, and history. In the 1960s, urban planner Lynch recognized the foremost impact of visual elements on the 'mental map' of a city's visitors and residents [24]. In recent years, multiple studies have attested the relationship between city visual characteristics and socio-economical statistics [27, 17, 3, 26, 6].

Thanks to pervasive city imagery datasets such as Google Street View, we are now equipped with tools to virtually explore cities and obtain similar knowledge. However, virtually walking the streets to generate insight is still slow, and tools to analyze the urban environment in summary from this imagery may hasten insight generation. As such, we present an application of city street-level image datasets for the systematic study of urban visual environments. We combine machine learning and data visualization to help consume, query, and analyze geographically-located image data.

We start with Lynch's theory of the five most important urban elements which impact mental maps of a place: paths, edges, nodes, landmarks, and districts [24]. Four of the five elements—paths, edges, nodes, and landmarks—denote individual objects which are simple to

- *Lezhi Li is with Uber Technologies. The work was completed at Harvard Graduate School of Design. lezhi.li@uber.com.*
- *James Tompkin is with Brown University. james_tompkin@brown.edu.*
- *Panagiotis Michalatos is with Harvard Graduate School of Design. pmichala@gsd.harvard.edu.*
- *Hanspeter Pfister is with Harvard Paulson School of Engineering and Applied Sciences. pfister@seas.harvard.edu.*

Manuscript received xx xxx. 201x; accepted xx xxx. 201x. Date of Publication xx xxx. 201x; date of current version xx xxx. 201x.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org.
Digital Object Identifier: xx.xxxx/TVCG.201x.xxxxxxx/

define. However, 'districts' is harder to define, as it refers to a collection of objects that share similar (potentially visual) qualities [1]. We postulate that the visual qualities which define districts can be found in geographically-embedded street view images, where images that are geo-spatially close and from the same district also tend to be close in their visual features. By learning what these features are via convolutional neural networks (CNNs), we aim to demark visual boundaries or 'perceptual neighborhoods' rather than administrative districts. As there are many potential factors that may emerge as we explore the different perceptual neighborhoods, we create tools to geographically analyze a hierarchical clustering of the learned visual features. With this, we provide tools both to help users evaluate the similarities and dissimilarities of urban districts within and across cities, and to help evaluate the effectiveness of the underlying machine learning.

Our work applies machine learning and visualization to help solve visual analytics problems in urban study. Such applications within the study of urban form and environmental psychology are uncommon because the discipline is largely taxonomy based. For example, researchers are often interested in discovering the distribution of classical and modern buildings, or the distribution of residential and industrial buildings. However, it is often difficult to acquire fine-grained ground-truth label data for this problem, which makes it hard to create a visually-representative model to classify images of the urban environment. We use hierarchical clustering to let 'classes' emerge from a pool of visual features learned on coarse administrative boundaries, which provides an alternative way to address this problem (Sec. 3.4).

The broader problem we set out to solve is hard because street view image data is often similar and so can be difficult to correctly classify, and because perceptual neighborhood boundaries are ill defined [37]. As such, assessing the quality of our results requires human judgment of the learned features (Sec. 3.4). For this task, we propose an interactive analysis of the relative importance of a hierarchy of learned visual features across clusterings of different data points. In this way, our work presents a more general tool to help understand the learned features in a convolutional neural network (Sec. 5).

We also contribute a more general visualization/interaction scheme to assist the comprehension of geospatially-embedded hierarchical data. Existing approaches for exploring these data often do not allow comparison across both the data hierarchy and the geographical locations of the data points. In our scenario, this data hierarchy contains the semantically-meaningful visual feature space. Our approach allows users to keep track of both the geographical and semantic interpretations of the data (Sec. 4).

2 Related work

2.1 Computer vision and geo-tagged images

Computer vision and especially CNNs are powerful tools for analyzing city visual qualities due to its ability to comprehensively capture visible features in image data. Applications of computer vision analysis to geo-tagged datasets have been growing in popularity. Many existing works focus on finding the geo-locations of objects or scenes. For example, Zhou et al. [45] tried to recognize city identity by associating scene attributes with location coordinates, and were inspired by the SUN scene attributes dataset [29]. Zheng et al. [44] and Li et al. [21] work on recognizing landmarks in cities. Hays et al. [15] inferred the geographical coordinates of images by using hand-designed image features through a data-driven scene matching approach, and Lin et al. [22] combined satellite images with ground images to more accurately pin down the geographical coordinates of an image. Workman et al. [38] used PlacesNet, ImageNet features, and a trained support vector machine (SVM) with radial basis function kernels to identify places in San Francisco.

Our goal is different from these kinds of work, as we do not aim to localize images. Instead, we try to discover how visual similarity/dissimilarity might exist between neighborhoods both within and across cities. Some works have studied the topic of city visual identity and similarity detection. Doersch et al. [11] classified mid-level features with SVMs to discover relevant image patches that contain distinctive architectural elements, and looked into the distribution of those elements across regions. In comparison, we learn the features that are distinctive in hopes of better discovering what makes a neighborhood distinct. Further, we attempt to discover a hierarchy of 'perspectives' on neighborhoods via an unsupervised clustering approach on the learned features, to discover both visual and geo-spatial neighborhood patterns.

Other works try to infer unseen urban context from what is seen in an image: Naik et al. [27] correlated a safety index with scene attributes, and Khosla et al. [17] used the "distance to McDonald's" metric to imply the relationship between scene and general urban structures. Arietta et al. [3] developed a method to automatically identify and validate predictive relationships between the visual appearance of a city and its non-visual attributes. Our technique can also be used in this way, e.g., to predict house prices from neighborhood appearance.

2.2 Visual analytics of spatial data

Urban analysis is an application domain frequently explored by the visual analytics community in recent years [43]. While some works in this field has been functionality driven, like those with regards to transportation analysis (Wang et al. [36], Di Lorenzo et al. [10]), others try to infer patterns and information regarding urban places such as mobility pattern discovery (von Landesberger et al. [34], Wu et al. [39]) or place semantics discovery (Andrinko et al. [2], Yu et al. [42]).

The real world connection in urban analysis inevitably requires geographically-embedded data structures, which can be a challenging spatial and non-spatial visualization integration problem. Many researchers have designed approaches for various non-spatial data structures in this pairing. For example, Yang et al. [40] incorporated map and matrix charts and used a "call out" method to display point-to-point relationships of geolocations inside of matrices. Love et al. [23] addressed spatial data where each data point has multiple possible values under different circumstances. Here, individual data points are sliced and diced within specific spatial contexts. Guo et al. [14] visualized spatial multi-feature datasets through a self-organizing map clustering such that different combinations of feature values translate to a 2D space for geographic map coloring.

Our data representation shares similarities across these three approaches: data points are arranged into largely-contiguous geographic regions (neighborhoods) with a need to assess geographic relationships on a map (comparing different neighborhoods), and each data point has multiple values from the learned visual features which lie within a clustered hierarchical structure. Existing approaches typically fail to represent all of these elements at once: matrices do not show hierarchies; slicing and dicing fails to show relationships among different data points; and self-organizing maps lose geographic information.

Our approach attempts to allow exploration across this data representation by organizing extracted image features into a hierarchy. Traditionally, spatial proximity is used to represent clusters for human evaluators. For example, t-SNE [25] is designed specifically to project distance in feature space to distance in visual space. Edge bundling [16] also helps viewers identify clusters using reduced edge distance. However, since the spatial channel in our representation is constrained to represent geographic information, we use a coordinated view and the concepts of connection in tree diagrams and containment in circle diagrams to convey the idea of a cluster (Fig. 1). This is similar to Chang et al. [8], who first aggregate information to create "legible clusters" which provide continuous levels of abstraction of data, and then use coordinated matrix, parallel coordinate, and choropleth views to help preserve mental models of the city. Along with using street-level imagery, we differ by providing an explicit representation of an aggregated hierarchy for exploration.

3 Extracting visual features of urban places

Visual urban district analysis requires us to scalably obtain a useful representation of visual features from street view images. Our investiga-

[1] Paths: streets, sidewalks, trails, and other channels in which people travel. Edges: perceived boundaries such as walls, buildings, and shorelines. Nodes: focal points, intersections or loci. Landmarks: readily identifiable objects which serve as external reference points.

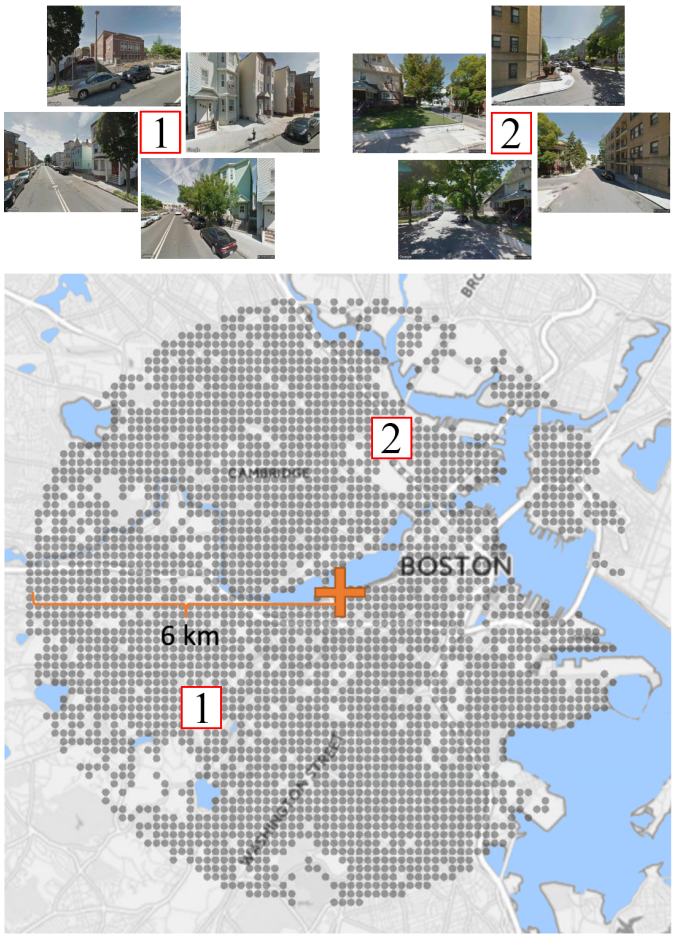

Fig. 2. Images are requested from coordinates within a 6 km radius covering the urban area, with each coordinate point represented by four images 90 degrees apart. Grey circles are coordinate sample points. As typically only road areas can be sampled for imagery, there are missing data, e.g., in parks.

tion had four stages: First, we collected street-level image data through the Google Street View API. We also acquired neighborhood boundary data from the Zillow Neighborhood Boundaries (Sec. 3.1). Next, we computed summary appearances across a city using semantic segmentation, which helps identify greenery, open spaces, and building color (Sec. 3.2). Then, we trained a convolutional neural network to classify the street view images according to the neighborhoods in which they are located. The purpose of this step is two-fold: to evaluate to what extent perceptual neighborhoods align with our labeled neighborhood boundaries, and as a way to learn a representation for neighborhood appearance (Sec. 3.3). Finally, we take the learned visual features from this network and cluster them hierarchically to produce a representation of the visual features which can be easily inspected using a visualization system (Sec. 3.4).

3.1 Data collection

Using the Google Street View API, we densely sampled four cities in the United States: Boston, Chicago, New York, and San Francisco. These cities were chosen for presentation because they are 'famous': their appearances and their similarities/dissimilarities are widely known. For each city, we drew a 6 km radius circle covering the major urban area, and collected images from a grid of lat-long coordinates within that circle. The spacing of the grid points are as follows: 0.0015° apart in longitude, and ΔL° apart in latitude, where

$$\Delta L = \frac{0.0015}{cos(\frac{Lat_{city}}{180} * \pi)}$$

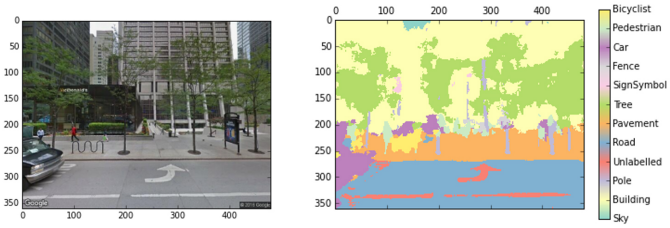

Fig. 3. Semantic segmentation of street scenes allows us to reduce the complexity of the image representation, and perform class-specific appearance modeling.

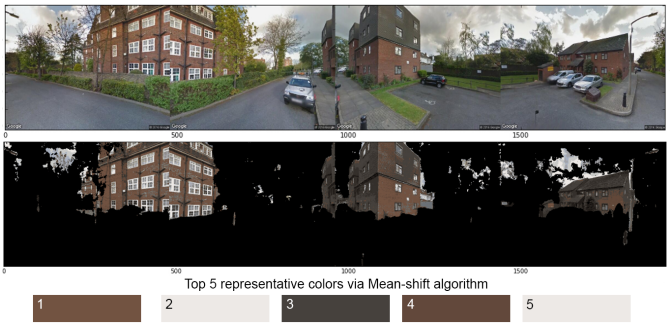

Fig. 4. Original street view images and their corresponding semantic building segmentation, along with the top five representative facade colors generated by the Mean-shift algorithm, Both the original image and the segmented image are concatenations of the four street view images taken from the same geographic coordinate.

where Lat_{city} is the city's latitude. The adjustment in latitude as compared to longitude ensures the earth-surface distance between adjacent 'rows' is the same as that of adjacent 'columns'. At each coordinate grid point, we collected 4 images with camera headings of h°, $(h+90)^{\circ}$, $(h+180)^{\circ}$, $(h+270)^{\circ}$ respectively, where h is a random number between 0 and 90 for different grid points. The randomness prevents unexpected bias when some parts of a city have a north-south street network while other parts have a diagonal-direction street network. Each image has 480 × 360 pixels (Fig. 2). After manually removing indoor and other invalid images from the obtained dataset, we had collected approximately 25,000 images from each of our four cities, totaling 100,000.

Additionally, we collect administrative district boundaries from the Zillow Neighborhood Boundaries database. This data is used as the ground truth labels for the neighborhood recognition task (Sec. 3.3.1).

3.2 Hand designed image features

Before proceeding with deep learning, we conducted an exploratory research on the effectiveness of human-identified image features in identifying various spatial patterns within and across cities. To discover broad spatial patterns within and across cities, we isolated specific semantic image segments and applied summary statistics across image pixels within each segment. We use SegNet [5], a trained CNN model which classifies pixels in a street view image into semantic classes such as trees, street signs, and cars (Fig. 3). With this, we can isolate pixels with classes that are most predictive to a task.

First, we counted all pixels belonging to the 'sky' and 'tree' classes and calculated the proportion of sky and tree area for every image. These values are averaged across the 4 images obtained for each lat-long coordinate (Sec. 3.1), and assigned as an attribute of each coordinate. These two attributes signify how 'open' a location is, e.g., high- or low-rise buildings, and how much greenery there is in an urban area.

Something slightly more complicated is needed to analyze buildings. If we chose to look at the 'dominant building facade color' as a major representative visual feature, then we must employ color quantization

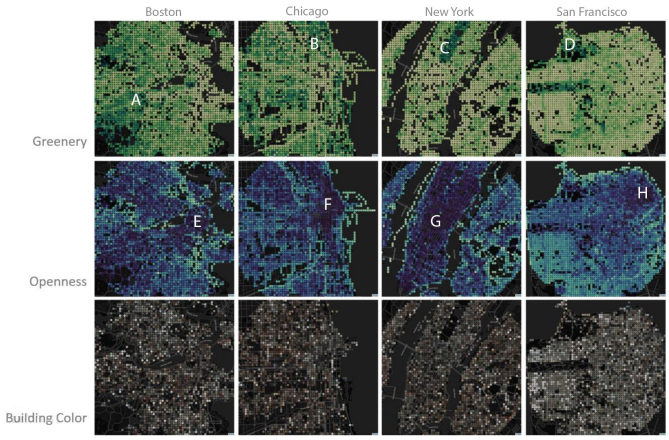

Fig. 5. *Top:* Visualization of image segments statistics reveals pronounced city patterns that conform to common sense: districts such as Chestnut Hill (A), Lincoln Park (B), Central Park (C), and Presidio (D) show high greenery amounts; city centers (E, F, G, H) present a lower level of sky area in images. *Bottom:* The Boston building color map. The discernible boundary between the white and the brown areas of Boston coincides with the boundaries of two neighborhoods: Back Bay (I) and South End (J).

techniques to select a representative building color. For instance, we might wish variations in red brick appearance to be averaged, and all window pane and frame colors to be ignored. Thus, we have a clustering problem. After experimenting with both K-Means [4] and Mean-shift [9] algorithms, we chose the latter because K-Means often resulted in balanced clusters and thus grayish representative colors, yet we wish to capture a distinctive saturated color. Mean-shift preserved the vividness of the original image as the representative color would remain largely invariant to small changes in the quantile size (Fig. 4).

Figure 5 shows each of the greenery, openness, and building color attributes mapped geographically. The patterns revealed in these maps agree with our personal perceptions of these cities. Even with just these summary statistics, we can begin to see some boundaries of perceptual neighborhoods. With this, we proceeded with deep learning. With automatically detected image features, what additional insights can we get from the dataset?

3.3 Learning and clustering boundary-trained features

Simpler approaches like the summary statistics in the previous section typically have insufficient descriptive power to separate sometimes very similar visual appearances. To evaluate in a more comprehensive way, many more visual features are needed from each image, and an automated process is required for feature extraction. With massive data and modern compute, we can do this by learning the visual feature representation using convolutional neural networks (CNN) [19, 13]. Multiple previous works compare the predictive power of hand-designed features (e.g., HOG, SIFT) to features learned from training CNNs on large image databases, and demonstrate that CNNs outperform on many tasks given sufficient training data [17, 12, 18]. This gives us a good indication that CNN-trained visual features are more appropriate for this research.

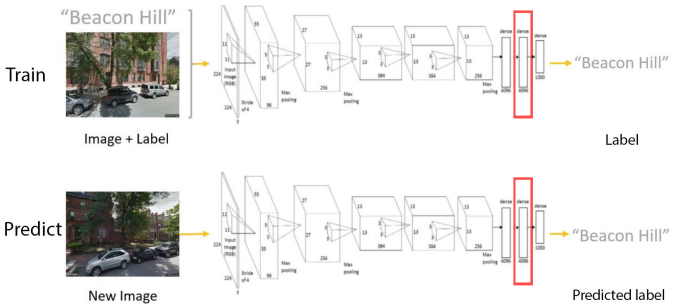

Fig. 6. Diagram of training and prediction process using AlexNet CNN (image based on original ImageNet paper [18], Fig. 2. *Red rectangles:* Fully-connected Layer 2, contains the 4096 features which we extracted as the feature representation of each image.

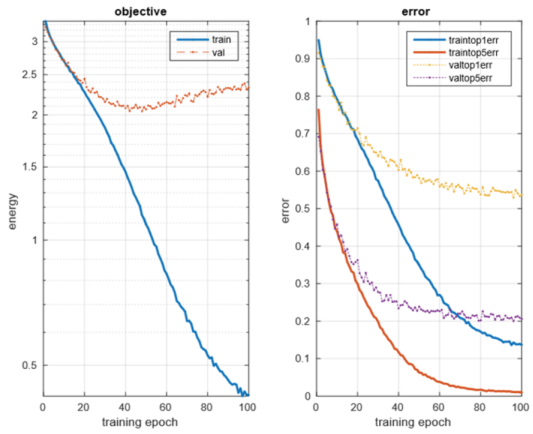

Fig. 7. Objective energy loss (*left*) and classification error rate (*right*) as CNN training progresses.

3.3.1 CNN training on neighborhoods

CNNs apply many layers of filters to images to transform them from pixel representations into feature representations. Then, we can use these features to make predictions, such as predicting the geographic location of an image from its appearance. To optimize an effective feature representation, CNNs require huge amounts of training data along with corresponding labels to give feedback to the network as to whether a certain prediction was correct or not (Fig. 6). In our case, the labels are which neighborhood the image comes from (Sec. 3.1).

We begin our learning process by splitting our dataset randomly into training, validation, and test sets in portions 4/6, 1/6, and 1/6 respectively. Then, we used the training/validation set to train the model to classify images according to their neighborhood names, using the AlexNet architecture [18] along with the correct number of class outputs in the final layer to correspond to the total number of neighborhoods in our study. The AlexNet architecture was chosen because it is well-known for giving good results on scene recognition tasks, and is relatively fast to train. We use the MatConvNet framework [33].

Fig. 7 shows the network training progress, where the model starts to over-fit after the 40th epoch. Final top 5 validation error lies around 25%, which shows the difficulty of the task. One important factor is the limitation of data for some small neighborhoods: the number of coordinates that fall into their boundaries may be insufficient for effective training, and denser sampling would produce mostly repetition.

Fig. 8. Images from five Boston neighborhoods (rows) with highest probability of being from the corresponding administrative districts.

3.3.2 Distinguishing districts

What has the training learned and does it makes sense to a human with knowledge of the area? Fig. 8 plots the top eight images across five areas of Boston for which the network is most confident about its classification prediction. This can be interpreted as answering the question 'which images of a neighborhood are most different from other neighborhoods?'. These images demonstrate some visual consistency within neighborhoods and visual distinction across neighborhoods, though this pattern in not as strong as in work which takes the same learning and classification approach but across entire cities [20].

What the algorithm cannot learn is also interesting. Intuitively, some neighborhoods might be too similar to each other to be distinguished; while other neighborhoods might have huge intra-class variance which can confuse the classifier. Either case could result in inadequate prediction power. For our study of the visual appearance of neighborhoods rather than the administrative district boundaries, situations where the model cannot distinguish between two districts are of equal importance to where the model performs well, as this inability to distinguish implies that the districts are visually similar.

Figure 9 shows a matrix of 'misclassified' cases among all Boston neighborhoods. The brightness of the cells indicate what percent of images in column categories (label data) were misclassified as row categories (predicted labels). The matrix can also be regarded as a node-link graph where nodes represent all the categories and links represent the levels of similarities between each pair of categories (i.e., an affinity matrix). We can apply a spectral clustering [28, 35] to the affinity matrix to cluster the nodes (neighborhoods) based on how strongly they are linked with each other.

Then, we sorted the order of neighborhoods in the matrix based on the result of this clustering process, to make neighborhoods that fall into the same cluster be near each other. Both columns and rows in Figure 9 follow this ordering. As we could tell from the red square, the most strongly-linked (having higher cell values) cluster of neighborhoods contains the North End, Downtown, the Leather District, Chinatown, and Back Bay (see A, B, C, D, and E in Fig. 10 for their geo-location). Those readers familiar with the city should find this result satisfying, as these areas constitute the city center (minus its large park), and are visually distinct from the farther-out suburb areas or the nearby seaport We varied the number of clusters and plot their results on maps (Fig. 10); as expected, districts with similar visual characteristics tends to be geographically close.

3.3.3 Limitations of neighborhood-based classification

From these findings, it is clear that simply applying a neighborhood-based classification approach does not make best use of the learned visual features. There are three reasons for this:

1. The granularity of our training label neighborhoods is insufficiently fine. Visual variance within the same neighborhood usually defies a single description.

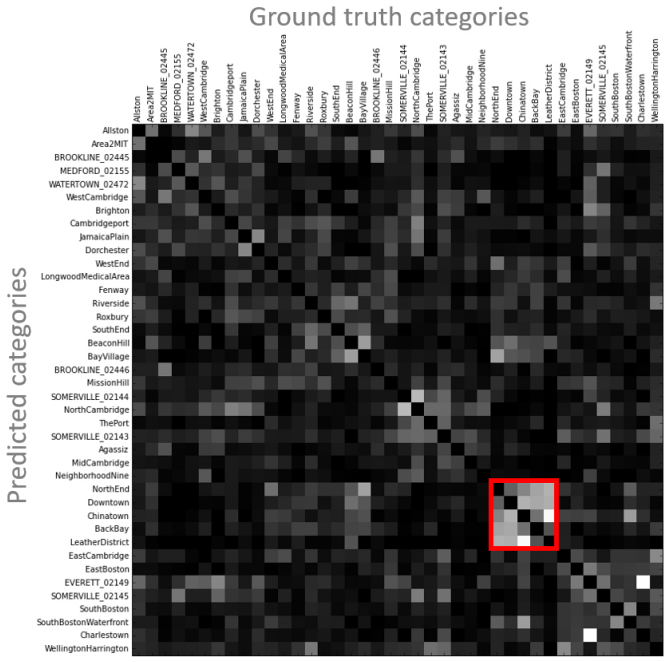

Fig. 9. Confusion matrix for areas of Boston. Brighter values indicate a higher rate of misclassification in the test set. Columns and rows were sorted by cluster id after applying spectral clustering (n_clusters=5) to all neighborhoods. *Red square:* The most strongly-linked cluster of the North End, Downtown, Chinatown, the Leather District, and Back Bay all share strong visual similarities.

2. 'Official' delineation of neighborhoods do not necessarily align with how neighborhood are perceived by humans; and often these boundaries are purely artificial and are prone to human manipulation. For instance, consider gerrymandering, a situation where politicians redistrict natural neighborhoods into new political boundaries for the purpose of getting more vote from the populations by region [1]). Even if official and perceptual neighborhood boundaries did not disagree with each other, it would still be interesting to learn the peculiarities where some parts of the city do not fall into their correspondent "categories". Simple classification does not allowed for these explorations.

3. The method cannot detect visual commonalities among different neighborhoods (or even different cities), which would provide an interesting pattern to indicate how prototypical design patterns have influence across borders. This might reflect important facts about the cities including zoning, cultural influence, or the trend of urban sprawl.

Therefore, there is need for a method which would be able to redefine the boundary of neighborhoods, purely from image features, using a bottom-up approach. Thus, we use our CNN in a different way: as a learned representation of visual features which we can use to interactively explore the relationships in the data. In the following sections, we demonstrate a machine learning techniques to let 'perceptual neighborhoods' emerge automatically from learned image features (Sec. 3.4), with an interactive visualization tool to help humans interpret and evaluate the results (Sec. 4).

3.4 Hierarchical clustering of visual features

In Sections 3.3.1 and 3.3.2, we investigated the classification prediction of to which neighborhood a certain image belonged, and found this method insufficient for many analysis tasks. Next, we will make better use of the learned visual features from the CNN's intermediate layers.

With the trained model from Section 3.3.1, we perform a forward pass on the test set of 17009 images. For each image, we extract the

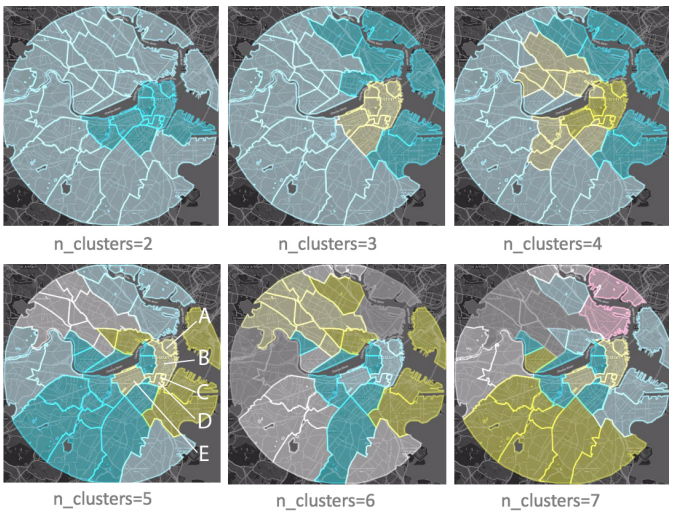

Fig. 10. Spectral clustering of neighborhood misclassification rates produces geographic clusters across the city. Bostonians past and present will notice the separation of industrial, commercial, and residential districts across these clusters. (A, B, C, D, E represent the 5 neighborhoods with the strongest linkage in Fig. 9: North End, Downtown, Leather District, Chinatown, and Back Bay, respectively.

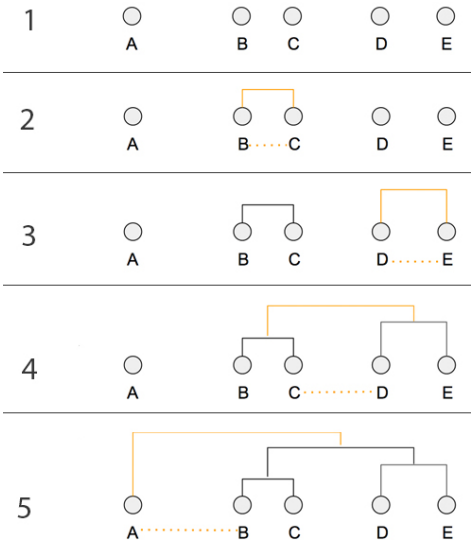

Fig. 11. Hierarchy-constructing processs in Agglomerative Clustering. Step 1 shows a collection of data points with different distances from each other in a one-dimensional space. From Step 2 to 5, the algorithm recursively finds the pair of nearest data points (indicated by the orange dotted line in each step) and merged them into a sub-tree. If one or both of the data points pair belongs to existing sub-trees, the sub-trees are merged with each other. Decision threshold increases with the steps. In the end, all data points become part of the adjacency hierarchy.

output from the second fully-connected layer, which is just before the final prediction layer of the neural network (indicated in Fig. 6 with red rectangles). This is represented as a 4096 dimensional vector for each image. Each vector element is either a positive number or zero, the value of which indicates the level of discernability of a specific visual feature as 'seen' by the neural network in that image. While these features can be inspected as to what they contain [41], at this depth in the network they are 'unnamed' and are often inconspicuous in their visual meaning.

As such, inspecting spatial distributions of these features in a similar way to in Figure 5 becomes difficult. Instead, we draw on clustering to both highlight pronounced variances and to reduce the dimensionality of the data being presented to a human for inspection. Clustering these 4096 x 17009 data is a challenge. We found it difficult to apply K-Means, DB Scan, or Spectral Clustering (RBF kernel) to this data, as setting parameters to produce a useful result was hard. Instead, we applied agglomerative clustering with Ward linkage and Euclidean affinity [30]—Ward linkage minimizes the variances of clusters being merged. This allows the user to inspect and vary the final clustering result to their desire by changing the level of agglomeration.

This algorithm iteratively merges the most similar data samples into clusters. Each iteration creates a new level of hierarchy, where the more dissimilar two clusters are, the later in the interative process they are merged (Fig. 11). After clustering, we can 'cut' this tree hierarchy at any desired threshold of similarity, to receive any number of clusters (top row, Fig. 12).

This is a desirable property for our task because, when looking at visual characteristics of urban areas, researchers are interested in both the different types of urban scenes (categorical) and how different they are from each other (continuous). For example, residential areas built at different time periods might look different in style (continuous), but are very different from an industrial area (categorical). As such, measures for visual similarity should be able to be categorized at different continuous values. Hierarchical clustering provides this function.

How can we evaluate that this produces meaningful results? The evaluation dilemma here is the absence of ground truth. Indeed, the whole point of developing this bottom-up clustering algorithm is to challenge the existing boundary system. Thus, the best we can do to measure success is to include a human in the loop and allow both their knowledge of the geography and their perception of image similarities to judge the outcome. Given this, interactive visualization is imperative.

4 Interactive Visualization

Designing an interactive visualization for the task of exploring these learned hierarchical feature representations requires combining geo-spatial data with our similarity hierarchy of visual features. Common methods for displaying tree-structured data do not preserve spatial relationships between data points; yet, this is core to our geographic analysis task. Further, our bottom-up hierarchical clustering approach provides the ability to observe how neighborhood boundaries change as the hierarchy is explored, and cutting this hierarchy provides the basis for exploring different perspectives on the visual features and challenging existing neighborhood boundaries.

However, hierarchical tree data can be difficult to comprehend when we wish to make sense of categories within the tree across different cutting thresholds:

- How might we allow users to vary the cutting threshold up to their desired granularity, and still convey a bigger picture of the degree of similarity/dissimilarity over the entire data set?
- How might we explain the phenomenon that, as the threshold varies, child and parent categories appear and disappear, and the assignment of a single data point varies?
- How might we express parent-children relationships among categories, especially when the positions of data points are geo-spatially confined?

Therefore, we developed a interactive visualization tool for analyzing this data representation, using D3.js [7]. The goal is for a human interacting with the interface, for instance, an urban planner, to be able to use both their knowledge of the geography and their perception of the similarities in the learned visual features for analysis. This tool might also be useful for researchers who wish to assess the validity of their machine learning-based approaches when applied to geo-spatial image data.

4.1 Interface

Our solution uses three coordinated views to present the data representation: the hierarchy is shown by a dendrogram, the categories are shown by a circle packing, and the geo-locations are shown by a choropleth

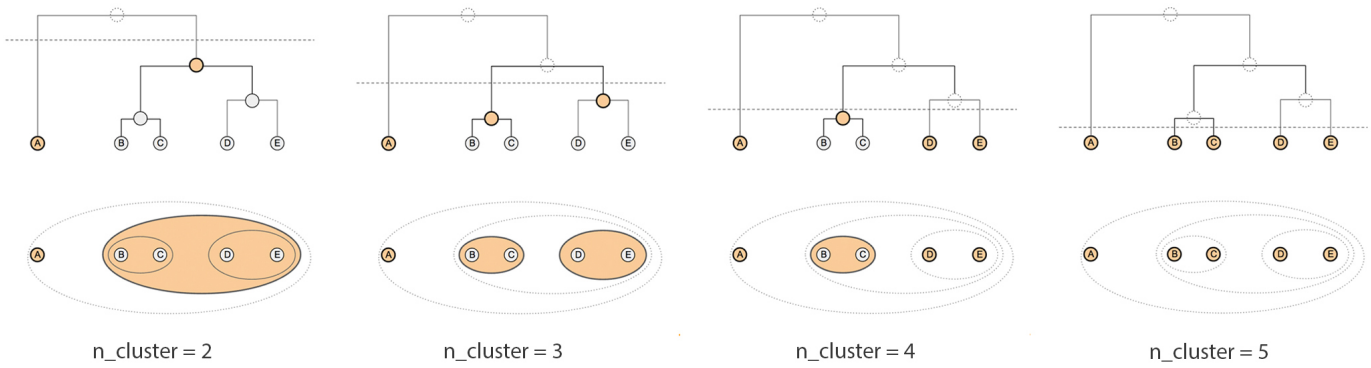

Fig. 12. Dendrogram and containment charts depicting two views of the same hierarchy across different decision thresholds. In sub-images 1 - 4, "current" categories under their respective decision thresholds (dotted line) are indicated in orange. The respective visual properties of the two charts help users to comprehend the notion of "categories" in the context of a hierarchy.

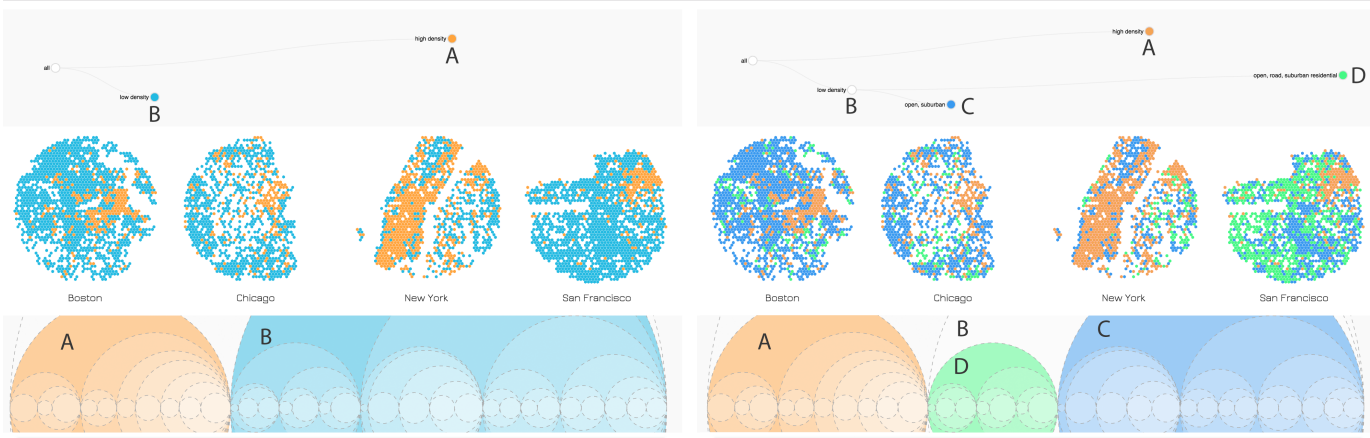

Fig. 13. Demonstration of a cluster exploration interaction. *Left:* Two visual clusters are observed, representing the top-level separation between high-density (orange, A) and low-density (blue, B) environments. E.G., Boston downtown (middle, left) has comparable density to most of Manhattan (middle, third from left). The hierarchy of potential clusterings is shown in the circle packing (bottom). *Right:* The user expands cluster B in the dendrogram (top), and clusters C (dark blue) and D (green) branch out. At the same time, the color for cluster B fades out, and only selected categories are highlighted with color. Unrelated clusters such as A remain stable in all three views, providing context.

map (Fig. 1). The dendrogram at the top provides users with the expert textual labels for the visual feature categories ('suburban', 'residential') which form from cutting the hierarchy at a particular level, along with an interface to expand each level into its child clusters. The choropleth map (upper middle) is the bridge between the hierarchy of visual clusters and geographic city locations in the city. The circle packing (lower middle) expresses clearly the containment or parent-child relationships between different visual features and their representative street view images. Colors across all three charts represent the same set of street view images, and so provides the conceptual link across views. This helps to relate real-world geo-locations to an abstract hierarchy of data which can be interactively cut at different thresholds to explore different clusterings. Once the hierarchy is formed, each level is assigned an intuitive label by an expert via data inspection, e.g., 'suburban', or 'residential', for easy user navigation. Figure 13 demonstrates a cluster exploration interaction, while Figure 14 demonstrates how individual clusters are queried for their corresponding street-level images.

While it might seem redundant to include both a dendrogram and a circle-packed representation of the visual characteristic hierarchy, they are intended to emphasize the continuous versus categorical aspects of this hierarchy, respectively. The branch analogy in the dendrogram is revealing for our application because the threshold level at which nodes are merged with their siblings is indicated by the horizontal position of the nodes (Fig. 13). For the circle-packing chart, the diameter of the circles corresponds to the number of images in the set. One confusing aspect with the dendrogram is that, since each node in the chart appears similarly as a small dot, users tend to comprehend each dot as a coexisting entity rather than as entities which appear or disappear depending on different thresholds. Circle containment clarifies this point visually by showing that one category is a part of another and can be further split into sub-categories. To highlight this point further, only those categories which are currently shown are colored, with the others of no fill color. The map view further accentuate this by showing colored data points that correspond to the current colored categories. Since no area on the map is ever not colored, it is easy to follow that unfilled categories are a 'state' of currently colored categories.

To summarize, the use of coordinated views provides the context and relations necessary to help users understand the correspondence between the real-world geo-locations and the hierarchical construct of learned visual features.

4.2 Usage scenario: model evaluation

One use for our visualization is to evaluate our machine learning approach of CNN and unsupervised clustering. Unlike many analytics schemes designed to address model accuracy issues, interactive visualizations for unsupervised training schemes are almost mandatory because the performance of a model cannot be verified against a ground truth labeling.

Researchers would be interested in, say, the biggest difference among all image samples that the model detects, as well as how much more similar area A is to B compared with A to C. When we take display and perception constraints into consideration, this conflict appears on

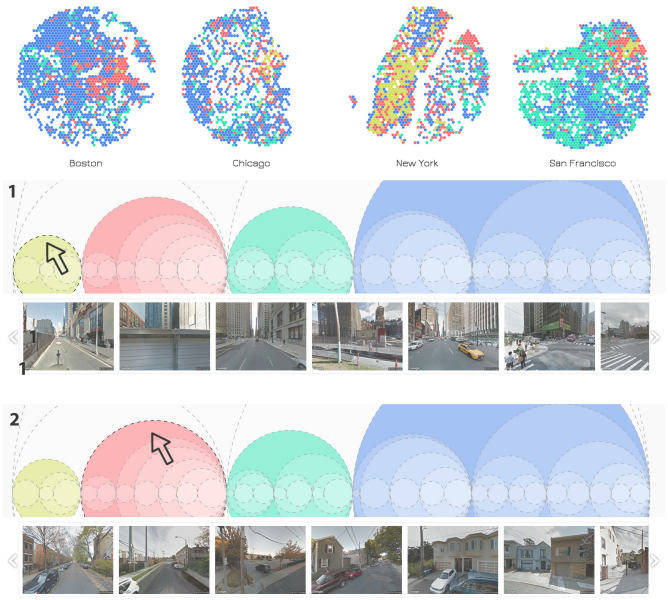

Fig. 14. Clicking the circle hierarchy categories fetches the street-level imagery which best represents that visual characteristic. In snapshots 1 (middle) and 2 (bottom), the user clicks on the largest yellow and red circles respectively, and discovers that the yellow circle represents images that contain more high-rise buildings than the red one. This is reflected in the map of New York (top, third from left), in which Manhattan is mostly yellow.

even more levels: it would be hard to display or to consume the large number of categories which result from a binary tree of visual features, where each image could be considered as a category in itself.

The solution was to use interactivity to control the display complexity, while allowing granularized comparison at the same time. Please see our video for a demonstration. The researcher would start with the root node which contains all images in the dataset, without any partitioning. Then, they click on the root node so that it splits into two sets that represent the most significant difference among the entire image set. Following this, they would observe the map for corresponding colored areas to evaluate whether the geo-space these clusters occupy actually have the most salient visual dissimilarity. If this outcome looks reasonable, it provides evidence to the researcher that the model is functioning as intended. Once the reseracher become interested in a more detailed portrait of data within a certain category, they click on that node to show the next child clusters on demand.

For example, in Figure 13, areas A 'high density' and B 'low density' are two child categories of the root note. As the user inquires into sub-categories of B 'low density', which contains children C 'open, suburban' and D 'open, road, suburban residential', the maps recolor to accommodate three categories. Area A remains the same color, which helps users to keep track of the higher-level structure of the hierarchy. The color scale, which is in proportion to the Y positions of tree chart nodes, also provides a reasonable separation of colors where the color distances indicate the visual similarity distances among all categories.

As users investigate the children of more and more categories, it would sometimes become difficult to discern the marginal difference on the map when an additional node is expanded or contracted. To avoid the visual confusion, the particular node that the user is operating on would first fade out, and then show the color(s) of the new categories (Figure 15). This emphasizing technique helps user attention to follow the part of the chart that has changed.

In general, the geo-spatial distribution of image clusters conforms to human intuition about the outcome of this image clustering approach (Fig. 13): the primary distinction captured is that between high-density and low-density regions in the cities. As we further partition the dataset, the cluster boundaries shown on the map tend to conform to human perception. When clicking on the circles to view images in a category (Fig. 14), it is typically not difficult to understand why they were grouped into the same set. Generally, at a local scale, visually similar images tend to be geo-spatially close, e.g., because high-density areas within one city tend to concentrate; at a scale across our four US cities, corresponding neighborhoods/regions are also visually close to each other, e.g., because high-density areas in Boston tend to look like high-density areas in Chicago.

4.3 Usage scenario: exploring in which area to live

Besides model evaluation, from the beginning of the design stage we wanted the visualization not only to be useful for researchers who work on the model themselves, but also for audiences who might not be automatically familiar with the methodology or model structure to begin with. Here we discuss one possible scenario as for how the visualization addresses both the model explanatory requirement as well as model evaluatory requirement.

Imagine you wish to gain a general understanding of the visual characteristics of an unfamiliar urban area. This demand occurs frequently when we explore where to live in a new city, and are wondering where to live. Despite the fact that services like Google Street View provides imagery of virtually any point in a city, there is no easy way to analyze this visual data in aggregate across neighborhoods. We might find the most convenient answers still lie in the perspective of a friend who lives there, or simply in visiting in person.

However, our learned visual features and visualization system can help, since it provides a quick and straightforward way for people to see this type of visual information at scale without requiring repeated point-by-point sampling on the map for street-level city imagery. If a user knows a particular area A that would be ideal for future housing, and wants to discover other similar areas to expand their selection, they are able to simply look for areas with the same color as A on the choropleth map. In this respect, training our representation (including neural network and clustering algorithm) serves as a data processing step to reduce the dimensionality of the image data into something that is easy to digest.

5 Discussion and Future Research

With the help of our tool, we were able to inspect how the neuron network sees cities, across different levels of differentiation and across different urban scenes. This CNN inspection helps the user to generate insight about the city. Our tool also helps generate perspective into how the CNN model structures visual information. For example, by applying feature agglomeration (an agglomerative clustering but on column space) and inspecting the results with our tool, we might be able to see the relative 'importance' of visual features, suggested by how deep these features are in the learned hierarchy, when the CNN is making sense of a highly homogeneous dataset like street view images.

One related question which we have not addressed is whether the fact that the CNN model is trained based on neighborhoods has anything to do with which visual features are learned. What if the model was trained to recognize cities, or to identify objects in scenes? Will the visual feature representation or the final clustering result be different? These questions are still open for future research.

Another question is how the learned visual features relate to latent non-visual characteristics of cities, e.g., house prices. To explain this point, we performed an experiment to test whether our features are related to house prices. We collected 500+ house price listings for Boston, assigned the values to the image points by proximity, and then split the dataset into training and testing as before. We learned a classifier for house prices using support vector regression with a radial-basis kernel [31], using two sets of features: one using just the geographic features of the data (latitude, longitude), and one using just the learned visual features from our model. Figure 16 compares the actual prices (X axis) to the predicted values (Y axis). The prediction using visual features ($R^2 = 0.4$) is better than the baseline model with only geographic coordinates. Considering visual appearance with learned features results in higher predictability of house prices.

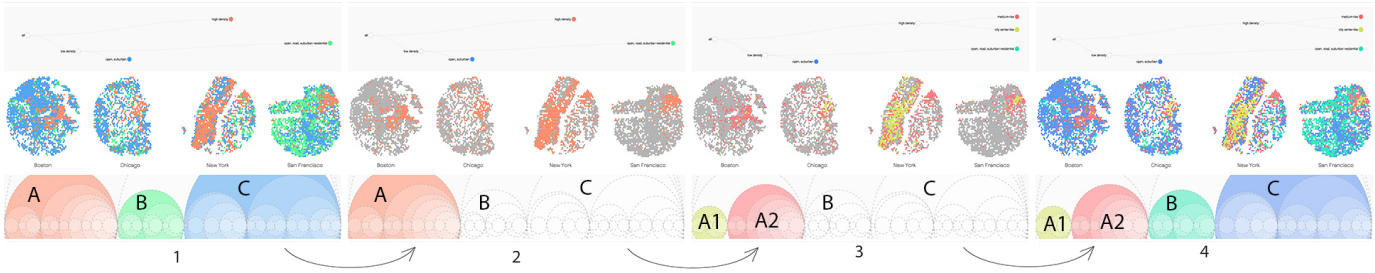

Fig. 15. When neighborhood A splits into A1 and A2, colors for B and C fade off, to allow users' attention to focus on the change of A.

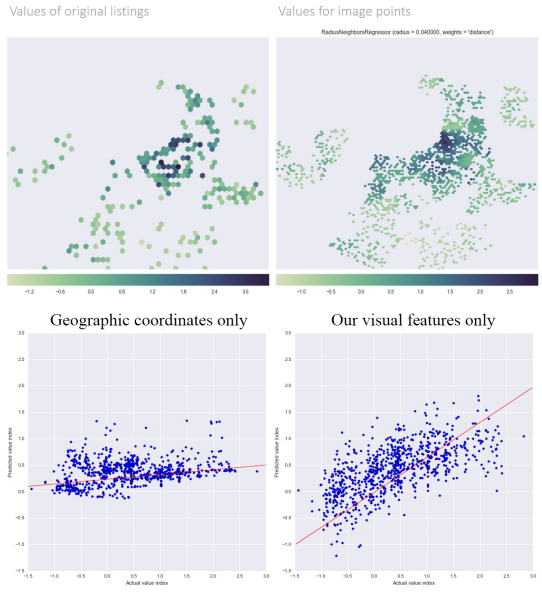

Fig. 16. *Top:* Average unit housing prices across Boston (left) are assigned to each image coordinate (right) according to their geo-spatial proximity to collected housing price data samples. *Bottom:* Actual versus predicted housing prices when only geographic coordinates are taken into consideration (left); actual versus predicted housing prices when using deep image features (right).

This provides evidence that our attempts to learn 'perceptual neighborhoods' actually captured something essential about the urban space, rather than just being a collection of similar images that happen to be located close to each other. We anticipate the relationship between the perception and the reality of cities to be worthwhile future research.

6 Conclusion

The bridge between visual features and internal meaning of the built environments has always been a principle topic in the discipline of architecture and urban design ("form follows function" [32]). Traditionally, the discipline of urban study has looked at patterns in a city from a top-down view, e.g., aerial imagery. Our proposed visual analytics methodology tests the idea that street-level visual properties also demonstrate significant patterns relevant to the study of the built environment. Our method automatically learns from imagery of the urban environment to reveal visual relationships in a fast and scalable way, where this information was previously difficult to digest when sampling street view imagery at individual locations. This is made possible with machine learning, which help reduce the dimensionality of the data, and with visualization, which addresses difficult geographically-embedded hierarchical datasets. With these approaches, we develop a software tool to query and explore the hierarchy of cities' perspective neighborhoods, and so help solidify Lynch's idea of the imageable city.

References

[1] Gerrymandering. `https://en.wikipedia.org/wiki/Gerrymandering`. Accessed: 2017-07-18. 2

[2] G. Andrienko, N. Andrienko, M. Mladenov, M. Mock, and C. Pölitz. Discovering bits of place histories from people's activity traces. In *Visual Analytics Science and Technology (VAST), 2010 IEEE Symposium on*, pages 59–66. IEEE, 2010. 2.2

[3] S. M. Arietta, A. A. Efros, R. Ramamoorthi, and M. Agrawala. City forensics: Using visual elements to predict non-visual city attributes. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):2624–2633, 2014. 1, 2.1

[4] D. Arthur and S. Vassilvitskii. k-means++: The advantages of careful seeding. In *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 1027–1035. Society for Industrial and Applied Mathematics, 2007. 3.2

[5] V. Badrinarayanan, A. Kendall, and R. Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *arXiv preprint arXiv:1511.00561*, 2015. 3.2

[6] A. J. Bency, S. Rallapalli, R. K. Ganti, M. Srivatsa, and B. Manjunath. Beyond spatial auto-regressive models: Predicting housing prices with satellite imagery. In *Applications of Computer Vision (WACV), 2017 IEEE Winter Conference on*, pages 320–329. IEEE, 2017. 1

[7] M. Bostock. D3.js. *Data Driven Documents*, 492, 2012. 4

[8] R. Chang, G. Wessel, R. Kosara, E. Sauda, and W. Ribarsky. Legible cities: Focus-dependent multi-resolution visualization of urban relationships. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1169–1175, 2007. 2.2

[9] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):603–619, 2002. 3.2

[10] G. Di Lorenzo, M. Sbodio, F. Calabrese, M. Berlingerio, F. Pinelli, and R. Nair. Allaboard: Visual exploration of cellphone mobility data to optimise public transport. *IEEE transactions on visualization and computer graphics*, 22(2):1036–1050, 2016. 2.2

[11] C. Doersch, S. Singh, A. Gupta, J. Sivic, and A. Efros. What makes paris look like paris? *ACM Transactions on Graphics*, 31(4), 2012. 2.1

[12] P. Fischer, A. Dosovitskiy, and T. Brox. Descriptor matching with convolutional neural networks: a comparison to sift. *arXiv preprint arXiv:1405.5769*, 2014. 3.3

[13] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`. 3.3

[14] D. Guo, J. Chen, A. M. MacEachren, and K. Liao. A visualization system for space-time and multivariate patterns (vis-stamp). *IEEE transactions on visualization and computer graphics*, 12(6):1461–1474, 2006. 2.2

[15] J. Hays and A. A. Efros. IM2GPS: Estimating geographic information from a single image. In *Computer Vision and Pattern Recognition, IEEE Conference on*, pages 1–8. IEEE, 2008. 2.1

[16] D. Holten. Hierarchical edge bundles: Visualization of adjacency relations in hierarchical data. *IEEE Transactions on visualization and computer graphics*, 12(5):741–748, 2006. 2.2

[17] A. Khosla, B. An An, J. J. Lim, and A. Torralba. Looking beyond the visible scene. In *Computer Vision and Pattern Recognition, IEEE Conference on*, pages 3710–3717, 2014. 1, 2.1, 3.3

[18] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012. 6, 3.3, 3.3.1

[19] Y. LeCun, Y. Bengio, et al. Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, 3361(10):1995, 1995. 3.3

[20] L. Li, K. Q. I. Lee, and J. Wei. Recognizing cities from google street view - computer vision analysis of urban visual identity. In *KDD '16 The 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2016. 3.3.2

[21] Y. Li, D. J. Crandall, and D. P. Huttenlocher. Landmark classification in large-scale image collections. In *Computer vision, 2009 IEEE 12th international conference on*, pages 1957–1964. IEEE, 2009. 2.1

[22] T.-Y. Lin, S. Belongie, and J. Hays. Cross-view image geolocalization. In *Computer Vision and Pattern Recognition, IEEE Conference on*, pages 891–898, 2013. 2.1

[23] A. L. Love, A. Pang, and D. L. Kao. Visualizing spatial multivalue data. *IEEE Computer Graphics and Applications*, 25(3):69–79, 2005. 2.2

[24] K. Lynch. *The Image of The City*. MIT Press, 1962. 1

[25] L. v. d. Maaten and G. Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(Nov):2579–2605, 2008. 2.2

[26] N. Naik, S. D. Kominers, R. Raskar, E. L. Glaeser, and C. A. Hidalgo. Computer vision uncovers predictors of physical urban change. *Proceedings of the National Academy of Sciences*, page 201619003, 2017. 1

[27] N. Naik, J. Philipoom, R. Raskar, and C. Hidalgo. Streetscore-predicting the perceived safety of one million streetscapes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 779–785, 2014. 1, 2.1

[28] A. Y. Ng, M. I. Jordan, Y. Weiss, et al. On spectral clustering: Analysis and an algorithm. In *Neural Information Processing Systems*, volume 14, pages 849–856, 2001. 3.3.2

[29] G. Patterson and J. Hays. SUN attribute database: Discovering, annotating, and recognizing scene attributes. In *Computer Vision and Pattern Recognition, IEEE Conference on*, pages 2751–2758. IEEE, 2012. 2.1

[30] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12(Oct):2825–2830, 2011. 3.4

[31] A. J. Smola and B. Schölkopf. A tutorial on support vector regression. *Statistics and computing*, 14(3):199–222, 2004. 5

[32] L. H. Sullivan. The tall office building artistically considered. *Lippincott's Magazine*, 57(3):403–409, 1896. 6

[33] A. Vedaldi and K. Lenc. Matconvnet: Convolutional neural networks for matlab. In *Proceedings of the 23rd ACM international conference on Multimedia*, pages 689–692. ACM, 2015. 3.3.1

[34] T. von Landesberger, F. Brodkorb, P. Roskosch, N. Andrienko, G. Andrienko, and A. Kerren. Mobilitygraphs: Visual analysis of mass mobility dynamics via spatio-temporal graphs and clustering. *IEEE transactions on visualization and computer graphics*, 22(1):11–20, 2016. 2.2

[35] U. Von Luxburg. A tutorial on spectral clustering. *Statistics and computing*, 17(4):395–416, 2007. 3.3.2

[36] F. Wang, W. Chen, F. Wu, Y. Zhao, H. Hong, T. Gu, L. Wang, R. Liang, and H. Bao. A visual reasoning approach for data-driven transport assessment on urban roads. In *Visual Analytics Science and Technology (VAST), 2014 IEEE Conference on*, pages 103–112. IEEE, 2014. 2.2

[37] A. Woodruff and T. Wallace. Bostonography. 1

[38] S. Workman and N. Jacobs. On the location dependence of convolutional neural network features. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 70–78, 2015. 2.1

[39] W. Wu, J. Xu, H. Zeng, Y. Zheng, H. Qu, B. Ni, M. Yuan, and L. M. Ni. Telcovis: Visual exploration of co-occurrence in urban human mobility based on telco data. *IEEE transactions on visualization and computer graphics*, 22(1):935–944, 2016. 2.2

[40] Y. Yang, T. Dwyer, S. Goodwin, and K. Marriott. Many-to-many geographically-embedded flow visualisation: An evaluation. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):411–420, 2017. 2.2

[41] J. Yosinski, J. Clune, A. Nguyen, T. Fuchs, and H. Lipson. Understanding neural networks through deep visualization. *arXiv preprint arXiv:1506.06579*, 2015. 3.4

[42] L. Yu, W. Wu, X. Li, G. Li, W. S. Ng, S.-K. Ng, Z. Huang, A. Arunan, and H. M. Watt. iviztrans: Interactive visual learning for home and work place detection from massive public transportation data. In *Visual Analytics Science and Technology (VAST), 2015 IEEE Conference on*, pages 49–56. IEEE, 2015. 2.2

[43] Y. Zheng, W. Wu, Y. Chen, H. Qu, and L. M. Ni. Visual analytics in urban computing: An overview. *IEEE Transactions on Big Data*, 2(3):276–296, 2016. 2.2

[44] Y.-T. Zheng, M. Zhao, Y. Song, H. Adam, U. Buddemeier, A. Bissacco, F. Brucher, T.-S. Chua, and H. Neven. Tour the world: Building a web-scale landmark recognition engine. In *Computer Vision and Pattern Recognition, IEEE Conference on*, pages 1085–1092. IEEE, 2009. 2.1

[45] B. Zhou, L. Liu, A. Oliva, and A. Torralba. Recognizing city identity via attribute analysis of geo-tagged images. In *European Conference on Computer Vision*, pages 519–534. Springer, 2014. 2.1