buildings

MDPI

Article

Robust Building Identification from Street Views Using Deep Convolutional Neural Networks

Robin Roussel [1,2,*], Sam Jacoby [1] and Ali Asadipour [2]

[1] School of Architecture, Royal College of Art, London SW7 2EU, UK
[2] Computer Science Research Centre, Royal College of Art, London SW11 4NL, UK
* Correspondence: robin.roussel@rca.ac.uk

Abstract: Street view imagery (SVI) is a rich source of information for architectural and urban analysis using computer vision techniques, but its integration with other building-level data sources requires an additional step of visual building identification. This step is particularly challenging in architecturally homogeneous, dense residential streets featuring narrow buildings, due to a combination of SVI geolocation errors and occlusions that significantly increase the risk of confusing a building with its neighboring buildings. This paper introduces a robust deep learning-based method to identify buildings across multiple street views taken at different angles and times, using global optimization to correct the position and orientation of street view panoramas relative to their surrounding building footprints. Evaluating the method on a dataset of 2000 street views shows that its identification accuracy (88%) outperforms previous deep learning-based methods (79%), while methods solely relying on geometric parameters correctly show the intended building less than 50% of the time. These results indicate that previous identification methods lack robustness to panorama pose errors when buildings are narrow, densely packed, and subject to occlusions, while collecting multiple views per building can be leveraged to increase the robustness of visual identification by ensuring that building views are consistent.

Keywords: building identification; street view imagery; CNN

Citation: Roussel, R.; Jacoby, S.; Asadipour, A. Robust Building Identification from Street Views Using Deep Convolutional Neural Networks. *Buildings* **2024**, *14*, 578. https://doi.org/10.3390/buildings14030578

Academic Editors: Pin-Chao Liao, Xiaowei Luo and Ting-Kwei Wang

Received: 11 December 2023
Revised: 9 February 2024
Accepted: 18 February 2024
Published: 21 February 2024

1. Introduction

This paper introduces a new visual identification method that links georeferenced building footprints to building elevations extracted from street view imagery (SVI). The goal is to improve a key step of data integration pipelines involved in building-level multimodal analysis of urban environments. This study focuses on architecturally homogeneous, dense residential streets featuring narrow buildings with a high level of repetition (terraced houses)—a common urban condition that is particularly challenging for previous visual identification methods because the location and orientation errors associated with street view panoramas can result in adjacent buildings being confused with each other. The key takeaway of this paper is that globally aligning panorama poses with building footprints using building elevations, detected and grouped across panoramas, significantly reduces identification errors, as demonstrated by a quantitative comparison with previous identification methods on a manually verified dataset collected in London, United Kingdom.

Panoramic street-level imagery is now an established source of data for many applications, including in health [1], environmental [2], and urban [3–5] research. In particular, the combination of geographic information systems (GIS) with SVI has been effective at enhancing urban analysis ranging from land use classification to property valuation, qualitative perception, and change detection [6]. These urban studies operate at various scales and granularities: while research on visual perception [7] and urban form [8] tends to consider elements such as building blocks and greenery as continuous components within street views, research concerned with attributes such as building age [9] or value [10] focuses

on individual buildings as individual atomic units. This discrete representation of the city involves an additional challenging step in the data processing pipeline: the visual identification of a large number of buildings within a set of images. This data integration step is key to unlock the potential of many works that focus on the computer vision side of building analysis, including change detection [11,12], façade segmentation [13,14], or urban reconstruction [15]. Indeed, most existing building image datasets are anonymized, with the exception of landmark recognition datasets (e.g., Google Landmarks Dataset v2 [16]), which by definition are not representative of the city as a whole. The need for visual identification goes beyond buildings: previous works have explored the automated auditing and inventory of various urban objects using SVI [17], including street signs [18] and trees [19,20].

In this paper, visual identification involves the mapping of a specific pixel region in a street view panorama to a building identifier, which can then be used to query other databases (such as planning applications, energy performance certificates, or transaction history). This mapping is performed via the proxy of a georeferenced building representation whose components are linked to an external identifier, such as a 2D map of footprints (possibly augmented with elevation data) or a 3D mesh. However, both SVI and geometric models spatially approximate the physical built environment, the former mostly because of panorama location and orientation errors and the latter mostly because of geometric simplifications [21]. Therefore, one of the main challenges of visual identification is the misalignment between SVI and the georeferenced model. Its impact is significant: previous works have reported an average alignment error of about 3 m [18,22], which is enough to have the intended building elevation lie mostly outside the view window computed using the building footprint, while parts of adjacent buildings become visible [23]. In a recent work, Zou and Wang [24] reported that filtering out street view images that do not show enough of the intended building elevation eliminated 60% of their original dataset. This misalignment, when combined with repetitive buildings at a high density, as well as occlusions (due to trees or vehicles) or privacy blurring, results in a high incidence of perceptual aliasing, whereby buildings are confused with their neighbors. These issues may have a limited impact when data derived from visual analysis are aggregated over larger areas [2], but they have the potential to become critical in many building-specific applications where the automation of visual assessment is already being considered, such as mortgage decisions depending on valuation [10,25], insurance premiums based on estimated vulnerability to environmental risks [26–28], or the flagging of abandoned houses [24].

The above challenges are recurrent in the research literature around building identification. Following the development of large-scale SVI services as well as handheld devices featuring cameras and geolocation in the 2000s, researchers started investigating ways to exploit geotagged visual data for applications such as image-based localization or visual place recognition [29–34], augmented reality [21,35–37], and urban modeling [15,22,38,39]. Many of these early works acknowledged and often tackled issues arising from the relative inaccuracy of GPS positioning in urban settings [21,22,29–36,39]. In parallel, advances in computer vision and deep learning, particularly around image classification and regression, object detection, and semantic segmentation, improved pipelines and unlocked new methodologies for architectural and urban analysis [5]. However, the integration of visual and geospatial data remains a challenge. On one hand, datasets are siloed and lack interoperability [5]; on the other hand, models and algorithms that are directly imported from computer vision lack any explicit use of the contextual information surrounding geotagged visual data [40]. Moreover, the unique perspective provided by SVI comes with specific limitations such as occlusions, privacy blurring, and geometric distortions [3], hampering not only visual analysis but also integration with other data sources. Compared to earlier works on image-based localization and augmented reality, the consumers of SVI have expanded from individual users to large-scale data processing systems with stricter requirements in terms of their ability to consistently and efficiently identify urban structures across geographic areas and time periods. The growing recognition of both the usefulness and limitations of SVI for urban analysis has led to increasingly sophisticated approaches

relying on deep learning models to either filter out low quality views [24,41,42] or detect objects of interest in the view [26,27,43]. Moreover, explicit building identification from SVI has taken center stage in many recent works [44–47]. While these approaches address some of the challenges around building identification from SVI, several research gaps remain.

The first gap revolves around the lack of architectural and urban diversity in the data used to evaluate these methods. Across all three main types of approaches found in the literature (coordinate-based view clipping [24,41,42,48–54], feature-based alignment [30,33–36,44,55–57], and instance matching [26,28,43,45–47,58]), the buildings considered in previous works are spaced relatively far apart, visually distinct, or both. This ignores the specific challenges of the densely packed streets of narrow, visually similar buildings, where misalignments and occlusions can hide the intended building so much that an adjacent building appears more visually prominent. As a result, while this problem is acknowledged [23], there is a lack of research on identification methods that are robust to panorama pose errors whose magnitude is significant relative to the typical building face width. This limitation is compounded by a second research gap: most existing urban analysis pipelines only retrieve a single view per building face, and those that collect multiple views [43,51] do not check their visual consistency, potentially resulting in views of different buildings being assigned to the same building.

The objectives of this paper are three-fold: first, to propose a method to distinguish the views of the same building face from views of neighboring building faces, and efficiently group views accordingly, thus enabling consistent multi-view identification. Second, to increase the robustness of building identification by proposing an approach that specifically addresses panorama pose errors. Third, to evaluate this strategy against previous methods on a dataset that features challenging examples absent from previous works, specifically: narrow, densely packed, and visually similar buildings. To achieve these goals, this paper introduces a new pipeline to collect as many good-quality views of each building as possible using deep learning object detection, followed by a deep Siamese convolutional neural network that is trained to recognize different views of the same building while distinguishing between adjacent similar buildings. The model provides similarity scores between pairs of building views that are used to perform spectral clustering, ensuring that views of the same building elevation are efficiently grouped together. The resulting groups are used to consistently align panoramas with their surrounding building footprints via an efficiently implemented global optimization. This allows building elevations detected in each realigned panorama to be robustly matched to their corresponding footprint. Evaluating the method on a new annotated dataset (sampled from Google Street View in various neighborhoods of London, where narrow terraced houses are common) shows that it outperforms previous identification approaches.

In summary, this paper makes the following contributions:

- A method to group the views of the same building face across panoramas using a deep similarity estimation network and spectral clustering.
- A pipeline to achieve a robust multi-view building identification, by combining this grouping method with a fine-tuned building elevation detector and a global optimization that realigns panoramas with their surrounding building footprints.
- An evaluation of the identification performance that compares the method to previous approaches on a new challenging dataset.

2. Related Work

Linking geometric urban representations and ground-level imagery is a widespread task that cuts across various applications and research fields. Street views of buildings are associated with their 2D footprint or 3D model by studies on visual place recognition [30,33,34], augmented reality [21,35–37], urban modeling [22,39,57,59], urban planning (e.g., change detection [55] and land use [41,48,49,51]), urban studies (e.g., age estimation [9,58] and gentrification [23,50]), environmental risk and energy assessment [26,28,42,54], and real estate [10,25,60].

With such diverse contexts comes a diversity of names for the same task, including building identification [44,58] and building recognition [45], image-model registration [57], combining, linking or joining maps and SVI [47,48,56], as well as building-image retargeting [22] and mapping [6]. These terms are not without semantic overlap with other tasks: building "identification" and "retrieval" can also refer to the task of finding the most likely matches of an input picture in a pre-existing image database [29,31,37,61], which is usually called "re-identification" when applied to persons [62] or vehicles [63]. The kind of visual identification considered in this paper, however, matches building images with their geo-referenced footprints or 3D models without any image already attached to these buildings.

Previous research has approached this problem in three main ways: coordinate-based view clipping, feature-based alignment, and instance matching.

2.1. Coordinate-Based View Clipping

The coordinate-based view clipping method takes a building footprint, retrieves the closest panorama within a predefined radius, and uses their respective geographic coordinates to calculate the appropriate heading, pitch, and field-of-view parameters to generate a perspective view that should contain the building. This is typically the way images are retrieved using the API of SVI providers such as Google Street View. Coordinate-based view clipping is a very common method in fine-grained land use classification papers: Li et al. [48] and Zhang et al. [49] predicted urban planning use classes at building or parcel scale in New York City, Kang et al. [41] and Srivastava et al. [51] predict building function labels from OpenStreetMap in various North American and French cities, respectively, while Sharifi Noorian et al. [52] focused on mapping retail stores in Manhattan, and Yao et al. [53] studied job-housing patterns in Shenzhen.

Coordinate-based view clipping is the simplest identification method because it does not require any image analysis. Its main limitation, however, is that the building of interest might not be entirely visible (or even not at all), or several buildings might be visible without differentiating the intended one. This limitation is acknowledged in building-level papers, with Ilic et al. [50] noting that geolocation issues may create false positives for change detection or Lindenthal and Johnson [10] and Szcześniak et al. [54] mentioning that occlusions may impede valuation and energy assessments, respectively. It is partially addressed by Kang et al. [41], Zou and Wang [24], and Mayer et al. [42] using a CNN classifier (or sequence of CNNs) to filter out non-building images, but as noted by Zou and Wang, a significant portion of the dataset (60%) is discarded in the process. In contrast, the method presented in this paper gathers views from multiple angles per building and uses object detection to localize the buildings in each view.

2.2. Feature-Based Alignment

Feature-based alignment tackles panorama pose errors by finding correspondences between geometric features (such as corners, edges, and outlines) derived from a georeferenced model and those detected in the image. It essentially casts visual identification as an image-model registration problem. Approaches can be divided based on the dimensionality of the model: using a 2D map, Cham et al. [30], Chu et al. [33], and Yuan and Cheriyadat [56] all used classical computer vision filters to find building edges or corners in SVI, compute candidate correspondences with map features, and use these candidates to refine the camera pose estimation. Using a 2.5D map or 3D model, Karlekar et al. [35] used silhouette matching through shape context descriptors, Lee et al. [34] performed this alignment manually using a custom interface, while Taneja et al. [55,64], Arth et al. [36], Chu et al. [39], and Park et al. [57] used semantic segmentation to extract building outlines and aligned them with model-derived outlines. Ogawa and Aizawa [44] went one step further by combining semantic segmentation with image depth estimation to find the optimal alignment by computing the pixel-wise distance between this depth map and the one derived from a 3D model.

This type of visual identification is effective when accurate feature correspondences can be found, but its performance drops when features are lacking, either because of heavy occlusions [64], or because buildings are too dense and similar in shape, forming a uniform built mass [44]. This is precisely the case with the data used in this study, which are rich in terraced houses. In contrast, the optimization proposed in this paper improves the robustness of image-model registration by finding correspondences between semantic image regions across panoramas and ensuring that multiple views of the same building overlap with its footprint.

2.3. Instance Matching

Instance matching is a more recent approach that relies on both geometry and texture to locate and separate individual buildings. It uses deep learning to detect regions (as bounding boxes or pixels) corresponding to individual buildings within a panorama (or a derived perspective view). Each detected image region provides a viewing direction expressed in 2D map coordinates via camera transformations that is used to intersect a building footprint. Both Zhang et al. [45] and Khan and Salvaggio [46] applied a bounding box detector to create a visual index of buildings in several North American cities (with the latter using an additional filtering step based on semantic segmentation), while Ogawa et al. [47,58] applied an instance segmentation model to do the same in Kobe. In recent examples of instance matching for urban analytics, Wang et al. [26,28] also used instance segmentation on SVI from Central and South American cities to estimate their vulnerability to environmental risks, while Xu et al. [43] estimated building heights in China.

While this approach is more robust to image–model misalignment than coordinate-based view clipping, and more robust to partially hidden building edges than feature-based alignment, it still assumes that the misalignment is not so severe that an adjacent building could be erroneously matched, which is questionable in dense residential areas such as those on which this paper focuses. Other works do combine building detection and geometric alignment: Kelly et al. [22] introduced a hybrid method for urban reconstruction that fuses a building footprint map, a 3D mesh, and multiple façade images extracted from SVI using global optimization, while in a remote sensing context, Chen et al. [65] detected and found correspondences between large structures to align satellite imagery. However, in both cases, the data sources are different from those used by the method below: it does not require a 3D mesh, and it operates on street-level rather than remote sensing images, which have very different characteristics in terms of occlusions and perspective changes. More importantly, the critical problem tackled in this paper is perceptual aliasing, i.e., how to disambiguate adjacent similar-looking buildings in the presence of location uncertainty. The novelty of its approach lies in formulating a global optimization that combines the typical instance-model alignment error (between a bounding box and a building footprint) with an instance–instance alignment error (between the bounding boxes of the same building across different views). Doing so, this paper aims to combine the advantages of feature-based alignment, instance matching, and building image retrieval into a robust visual indexing method that identifies multiple views per building.

3. Method

3.1. Overview

In this work, the task of visual indexing from street views consists of linking each building face visible from the street with a set of pixel regions from nearby panoramas. Building footprints are represented as georeferenced polygons in a 2D map, equipped with a unique identifier and an estimate of the elevation of their highest point. Street view panoramas come with noisy location and orientation data (collectively referred to as "pose") provided by GPS and inertial sensors.

This paper uses the following definitions and notations (illustrated in Figure 1a).

- Building face F_i: the physical part of a building's envelope, such as the façade, whose components (i) are approximately parallel to a dominant plane; (ii) have a combined

length of at least $l_f = 3$ m; and (iii) are offset from each other by less than l_f. Examples are shown in Figure 1b. The centroid of F_i is noted M_i.

- Panorama P_j: 360-degree photo whose pixels map to spherical coordinates (polar and azimuthal angles) relative to the panorama center C_j. The set of indices of panoramas in which F_i potentially appears is noted as $\mathcal{J}_i$.
- View window W_{ij}: rectangular window, defined in world coordinates, through which rays connecting C_j to all visible parts of F_i must pass.
- Elevational street view (ESV) V_{ij}: image of building faces obtained by rendering a panorama P_j through the view window W_{ij}. While F_i is the "intended building face" of V_{ij}, it might not be visible, and adjacent faces might be visible. The set of elevational street views with intended face F_i is noted as $\mathcal{V}_i$.
- Building elevation E_{ijk}: projected view of a building face as visible in an elevational street view V_{ij}.
- Building elevation group $\mathcal{G}_{il}$: set of building elevations that correspond to the same building face in different elevational street views of F_i. This face can be F_i or an adjacent face.
- Building elevation ray R_{ijk}: ray from C_j through the center of E_{ijk} 's bounding box.

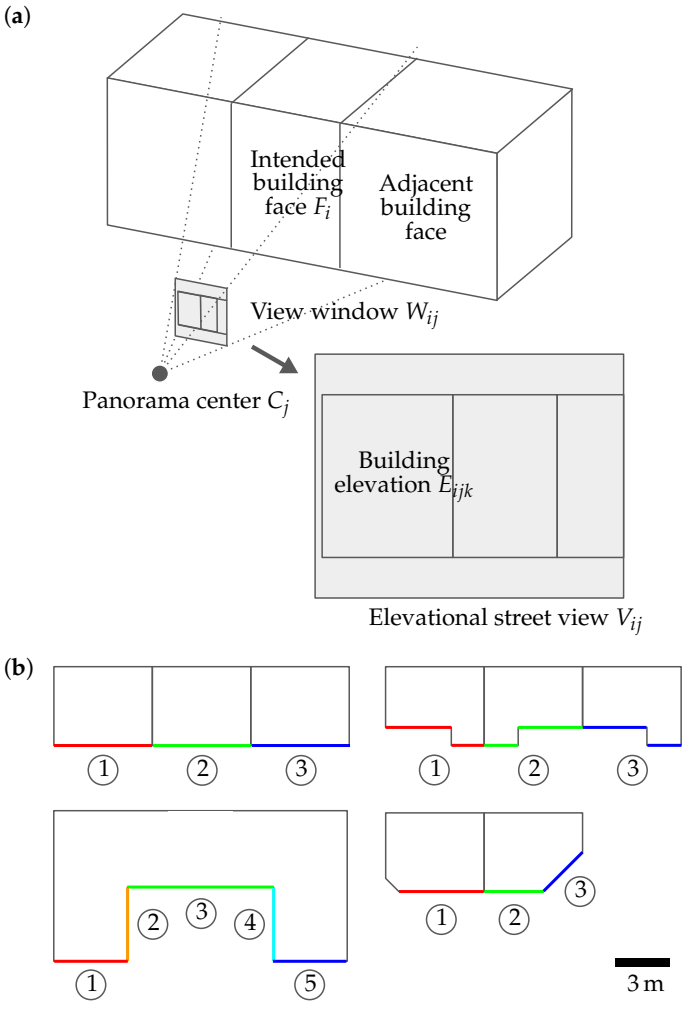

Figure 1. Concepts and definitions: (**a**) Illustration of the definitions and notations used in this paper; (**b**) Illustration of the concept of "building face" used in this paper. Four examples of building configurations, viewed from the top, where each number is associated with a single color corresponding to a single face.

The visual identification method proposed in this paper, henceforth called group-aligned matching, involves the following steps (illustrated in Figure 2):

1. Find the set of *exposed* building faces, where each face F_i potentially appears in a set of panoramas with indices $\mathcal{J}_i \neq \emptyset$. (Figure 2a and Section 3.2)
2. For each face F_i potentially visible in P_j, compute the view window W_{ij} and render the elevational street view V_{ij}. (Figure 2b and Section 3.3)
3. For each elevational street view V_{ij}, detect the visible building elevations $\{E_{ijk}\}$ that it contains (Figure 2b and Section 3.4).
4. Across all elevational street views $\mathcal{V}_i$ of each building face F_i, estimate the visual similarity between pairs of building elevations $\{E_{ijk}, E_{ij'k'} \mid (j, j') \in \mathcal{J}_i^2, j \neq j'\}$ (Figure 2b and Section 3.5).
5. Using this similarity metric, estimate building elevation groups $\{\mathcal{G}_{il}\}$ for each building face F_i (Section 3.6).
6. Using these building elevation groups, optimize all panorama poses to improve the alignment of all elevation rays $\{R_{ijk}\}$ with the exposed building faces (Figure 2c and Section 3.7).
7. For each elevational street view V_{ij}, obtain the building elevations whose ray R_{ijk} intersects the intended face F_i. If there is at least one, assign to F_i the building elevation whose ray is closest to M_i. (Figure 2d)

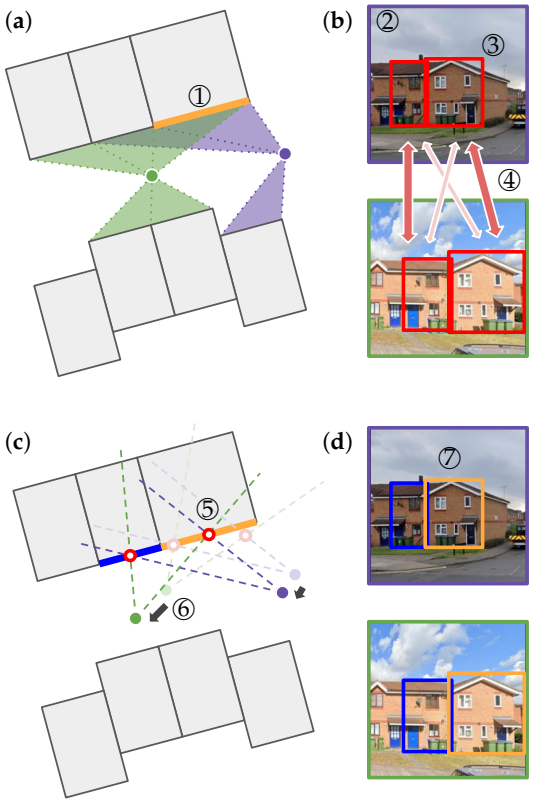

Figure 2. Overview of group-aligned matching. (**a**) Step 1: Exposed building faces are found using building footprints and the poses of nearby street view panoramas (one green, one purple), then a face is selected (thick orange line). (**b**) Step 2: Elevational street views are rendered. Step 3: Building elevations are detected (red boxes). Step 4: Their pairwise similarity is estimated (red arrows, shade/thickness reflects similarity). (**c**) Step 5: Building elevations are grouped (grouped ray intersections are shown in red). In this example, two groups are found, potentially matching two building faces (shown as thick lines, intended face in orange and adjacent face in blue). Step 6: The rays of each group are used to align panoramas with the building footprints (alignment shown with dark arrows). (**d**) Step 7: Building elevations are identified in each elevational street view if their aligned ray is closest to the centroid of the intended face (identified building elevation box in orange, adjacent building elevation box in blue).

For comparison with the identification approaches defined in Section 2, coordinate-based view clipping stops at step 2, while instance matching includes steps 1–3 and 7. Steps 4–6 constitute the main contribution of this paper.

3.2. Finding Exposed Building Faces

The first step of visual identification is to define and acquire the entities that street view elevations will be identified with. Specifically, the aim is to obtain a collection of line segments with the three following properties: they are indexed (i.e., they have a unique identifier that can be used to link them to other databases), georeferenced (i.e., located in a specific system of geographic coordinates), and each corresponds to a building face that can be seen from the street (see Figure 3). The starting point of this process is a 2D map of building footprints where each polygon represents a single building and is associated with at least one postal address and an estimate of the elevation (here, "elevation" means "height above sea level") of its highest point.

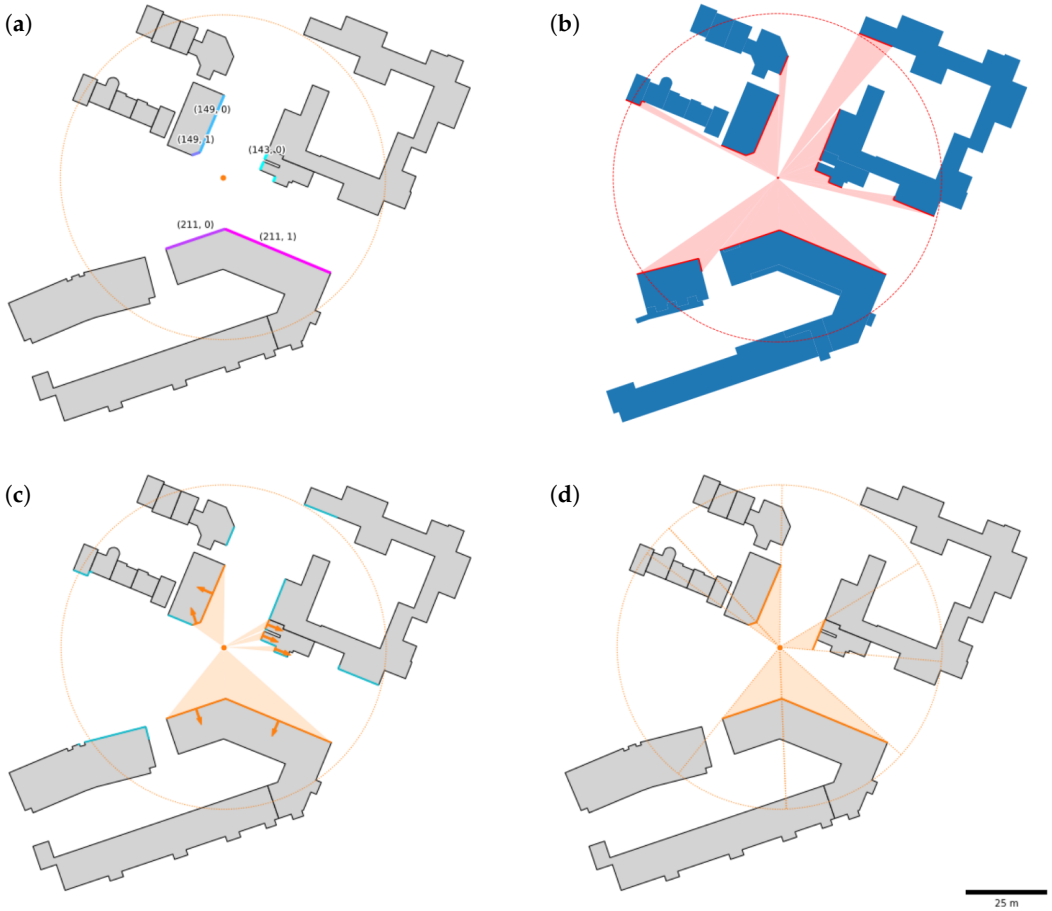

Figure 3. Computing exposed building faces and their respective view windows. (**a**) Polygon edges are grouped into indexed building faces (only some edges are shown). Each color corresponds to a single building face. (**b**) A visibility algorithm is used to determine the portions of building polygon edges (red lines) reachable by rays originating from the panorama center (red dot), at a maximum distance indicated by the red circle. (**c**) Building faces are kept if they satisfy viewing quality criteria, including a maximum angle between panorama rays and the normal vector to each building face (orange arrows). Valid and excluded building faces are shown in orange and blue respectively. (**d**) View windows are computed for each remaining building face (orange line segments). The angular bounds of each view window are indicated by dotted orange lines originating from the panorama center (orange dot).

In order to visually match what is conventionally read in architectural terms as a building elevation, polygon edges need to be grouped into building faces, which are sequences of almost contiguous, approximately parallel walls (see Figure 3a). First, an edge is defined as significant if its length is at least l_f (defined in Section 3.1), and two edges are considered to have a similar orientation if the angle between them is less than a threshold α_e (taken as 10° in this work). Starting with an arbitrary edge and going along the polygon boundary, edges are aggregated as long as they are approximately parallel and are not interrupted by a significant edge with a different orientation. Each resulting edge group is then given a face index i that is used to reference the subsequent sets of elevational street views $\mathcal{V}_i$.

These faces, however, are only considered for further processing if they satisfy specific viewing quality criteria for at least one panorama. There are three such criteria, which are evaluated via the faces' coordinates and elevation attribute as well as the coordinates of nearby panoramas. Let F_i be a building face with centroid M_i and P_j a panorama with origin C_j. First, the visible proportion of F_i within P_j needs to exceed a length ratio of r_v. This "visibility", however, is purely geometric and obtained from an angular sweep algorithm [66] used to compute a visibility polygon from C_j (see Figure 3b). Any face that does not satisfy this condition is too heavily occluded by another building. The second criterion is the maximum angle α_n between the normal vector to F_i (pointing away from C_j) and the ray from C_j through M_i (taken as 45° in this work). Any face that exceeds this angle will be too distorted in the resulting elevational street view. The last criterion is the maximum absolute elevational difference between the base of F_i and P_j's camera height (taken as 2.4 m following Liang et al. [67]). This is a simple way to detect panoramas taken from overpasses or tunnels, which are unlikely to display the intended building face.

3.3. Rendering Elevational Street Views

The view window W_{ij} of any face F_i that satisfies the viewing quality criteria for P_j is then computed by taking the two bounding rays from C_j towards each extremal vertex of F_i (in terms of polar angle). These rays are intersected with the dominant plane of F_i going through the face's vertex that is closest to the camera. The resulting horizontal line segment and the vertical line segment going from the base of F_i to its maximal elevation define a narrow view window $\hat{W}_{ij}$. The view window W_{ij} is obtained by expanding $\hat{W}_{ij}$ in all four directions with a margin factor m_v. As demonstrated in Section 4.2.2, a margin factor of 1 (meaning that a margin equal to the width and height of $\hat{W}_{ij}$ is added to the left/right and top/bottom sides, respectively) ensures that views consistently include the entire elevation of their intended building face.

An elevational street view V_{ij} is obtained by transforming a region of a panorama P_j that is expected to contain an intended building face F_i. The resulting image approximates a perspective projection obtained from a pinhole camera. The transformation consists of sampling rays from C_j through a regular grid of points within the view window W_{ij} using a constant resolution (80 pt/m in this work), and interpolating the pixels in P_j closest to each ray. In that sense, it is akin to rendering. An additional aim is to make all views $\mathcal{V}_i$ of the same intended face F_i comparable for identification purposes, which involves projecting them onto a common image plane that is fronto-parallel to the dominant plane of F_i. In that sense, the transformation is also a form of rectification. The quality of the rectification in each image V_{ij}, however, depends on the alignment between the view window W_{ij} and the dominant plane of F_i, as well as the quality of P_j itself (which might present stitching artifacts). In this paper, it is assumed that W_{ij} is close enough to the dominant plane of F_i to allow visual identification.

3.4. Elevation Detection

Due to the necessary margins added to the narrow view windows, the number of building elevations fully included in each elevational street view is unknown: there might be none (because of occlusions not represented in a building map), one, or more (because the

elevations may be narrow and close together). Elevations need to be not only recognized but also individually localized within the image, which is the typical setup of object detection. The detector used has a Faster R-CNN architecture [68] with a ResNet-50-FPN backbone [69] pretrained on ImageNet. For each image, the model outputs a number of bounding boxes associated with a confidence probability.

The last two layers of the network are fine-tuned on a custom elevation dataset made up of 2000 elevational street views annotated with bounding boxes drawn around building elevations. The dataset is randomly split 90%/10% between training and test sets, and another 20% of the training set is randomly selected before each training run for validation and hyperparameter optimization, yielding the following values. The optimization is performed via mini-batch stochastic gradient descent (SGD) with a batch size of 2, a base learning rate of 10^{-2}, a momentum of 0.9, and no weight decay. The training is run for 12 epochs, with the learning rate decayed by a factor of 0.1 at the tenth epoch. To reduce overfitting, the training data are augmented using random horizontal flipping with 50% probability and random changes in brightness, contrast, and saturation with a magnitude of 0.5 and 50% probability.

3.5. Pairwise Similarity Estimation

Following similar problem settings found in different domains (e.g., person [62] or vehicle [63] recognition), the estimation of building elevation similarity is solved using a deep metric learning approach. The aim is to compute an image embedding such that the Euclidean distance between feature vectors is small for images of the same building and large otherwise. While alternative methods exist for image similarity estimation [70,71], notably using the SIFT [72] or SURF [73] local descriptors, early experiments with local feature matching suggested that this approach would be too sensitive to the significant amount of similar local features between adjacent building faces, due to the large degree of visual repetition found in each image of the dataset (identical windows, doors, etc.).

Deep metric learning only requires a dataset of positive and negative pairs of images, which is produced by annotating pairs generated from the elevation detection dataset (see Section 4.1.2). The resulting elevation similarity dataset is randomly split between 90% of building faces used for training and 10% for testing. Splitting by building faces rather than elevation pairs ensures that the test set contains faces never seen before by the network and, in practice, the resulting proportion of elevation pairs used for training is almost identical (89.8%). As with the detection data, another 20% of training elevation pairs is randomly split before each training run for validation and hyperparameter optimization. The images provided to the network during training, testing, and inference are obtained by cropping around each elevation box with a 300 px margin on each side to provide context, resizing to a 256 px square, and normalizing the per-channel average and standard deviation using values precomputed on the training set. Lastly, a random horizontal flip is applied with a 50% probability to each pair. It is not applied independently to each image because knowing the specific layout of an elevation is an important clue to distinguish between adjacent building faces.

The elevation similarity network has a Siamese architecture whose backbone is similar to the elevation detector (ResNet-50-FPN pretrained on ImageNet). The last spatial pooling layer, however, is replaced by a channel pooling layer implemented as a 1×1 2D convolution. This change trades the backbone's translation invariance (less critical because the detection step ensures that each elevation is centered in the image) for the ability to represent the spatial layout of components within an elevation. The head of the network as well as the training loss are also quite specific in that the network is built to output a similarity score between 0 and 1 (akin to a match probability) rather than a distance, and the loss is the binary cross entropy between the prediction and the label. Using a classification loss in a deep metric learning setting is a relatively recent approach that has provided good results [74,75], alleviating the need for complex sampling strategies. In this paper, the similarity is calculated by taking the element-wise squared distance between

the feature vectors and applying a single fully connected layer followed by a sigmoid. The fully connected layer allows the network to focus on the parts of the image that matter most for the similarity evaluation.

Following hyperparameter optimization, the head and last four layers of the backbone are trained for 40 epochs using mini-batch SGD with a batch size of 32, a base learning rate of 3×10^{-3}, a momentum of 0.85, and a weight decay of 10^{-3}. The learning rate schedule includes a linear warm-up for the first 200 optimizer steps (starting with a factor of 10^{-1}) in order to stabilize the training. The training data are only augmented using random horizontal flipping with 50% probability as other augmentations were found to have a negative effect.

3.6. Elevation Grouping

The similarity scores between elevations $\{E_{ijk}\}$ sharing the same intended building face F_i are used to create the building elevations groups $\{\mathcal{G}_{il}\}$. Spectral clustering refers to a class of algorithms that can efficiently compute such a grouping from a similarity matrix by computing a low-dimensional embedding of the matrix (typically taking the first few eigenvectors of its Laplacian) and clustering the corresponding vectors (e.g., using k-means clustering) [76]. This method requires the number of groups k to be defined, which is unknown since the elevational views can contain several buildings. This number is determined by running successive rounds of spectral clustering with increasing values of k. The best k maximizes the mean silhouette coefficient of samples, which measures the quality of clustering [77]. To reduce the number of rounds, the possible values of k are bounded by setting the minimum number of groups as the highest number of elevation boxes in a single image, and the maximum as the total number of boxes minus one (assuming that at least two boxes can be clustered, which is checked before grouping by testing that at least one pair has a similarity score above 0.5).

3.7. Panorama Alignment

Every step so far has assumed that building footprints and panoramas are misaligned. Feature matching methods (as described in Section 2) show that panorama poses can be refined by finding correspondences between (i) panoramas and footprints and (ii) between groups of panoramas. The novelty of the present method lies in matching semantic image regions (i.e., building elevation boxes) rather than geometric image features. The alignment task is formulated as a geometric optimization with two objectives (illustrated in Figure 4): (i) obtain each building elevation ray R_{ijk} closer to the centroid of its nearest building face (independently of the intended face F_i); and (ii) obtain the intersection of elevation rays from each group $\mathcal{G}_{il}$ closer to the dominant plane of their intended face F_i. Formally, the optimization aims to find

$$\arg\min_{\Theta} [O_e(\Theta) + O_g(\Theta)]$$

with

$$O_e(\Theta) = \sum_{i,j,k} \min_{i' \in \mathcal{N}_j} d(M_{i'}, R_{ijk}(\Theta_j))$$

$$O_g(\Theta) = \sum_{i,l} d(Q_{il}(\Theta_{il}), F_i).$$

Here, O_e is the building elevation objective and O_g is the building elevation group objective. The vector Θ_j represents the pose parameters of panorama P_j, while Θ is the concatenation of all Θ_j and Θ_{il} is the concatenation of the Θ_j of all panoramas associated with $\mathcal{G}_{il}$. The set $\mathcal{N}_j$ contains the indices of all building faces near the panorama P_j (within a radius of 30 m and whose normal vector points towards P_j). The point Q_{il} is the least-squares intersection of all building elevation rays in the group $\mathcal{G}_{il}$. Lastly, the function d is the point-line distance.

The building elevation objective O_e increases intra-panorama consistency by simultaneously aligning all building elevations in each panorama with their surrounding building faces, while the building elevation group objective O_g increases inter-panorama consistency by aligning all grouped building elevations. Moreover, formulating O_e in terms of distance to the closest face simplifies the optimization by avoiding explicit assignments that would introduce discrete variables. Although the optimization is global, panorama poses are loosely coupled because moving one panorama only affects the building elevation groups it is associated with. Each entry of the Jacobian vector, which corresponds to the derivative with regard to a pose parameter of a single panorama P_j, is efficiently determined by only recomputing the ray intersections and distances with which this panorama is associated. Because of the high likelihood of local minima, the optimization is solved using the basin-hopping algorithm [78] with L-BFGS-B as its local minimizer [79]. Panorama poses are bounded within 3 m on each side of the initial location parameters and 5° around the initial heading parameter. The algorithm is found to converge within 20 global iterations with a local minimization tolerance of 10^{-3}.

Figure 4 shows an example of optimization results. In the left column, the top-left red ray and the bottom-right green ray, for example, initially intersect the wrong face. After optimization, rays go through the correct building face centroids and group intersections lie on building faces.

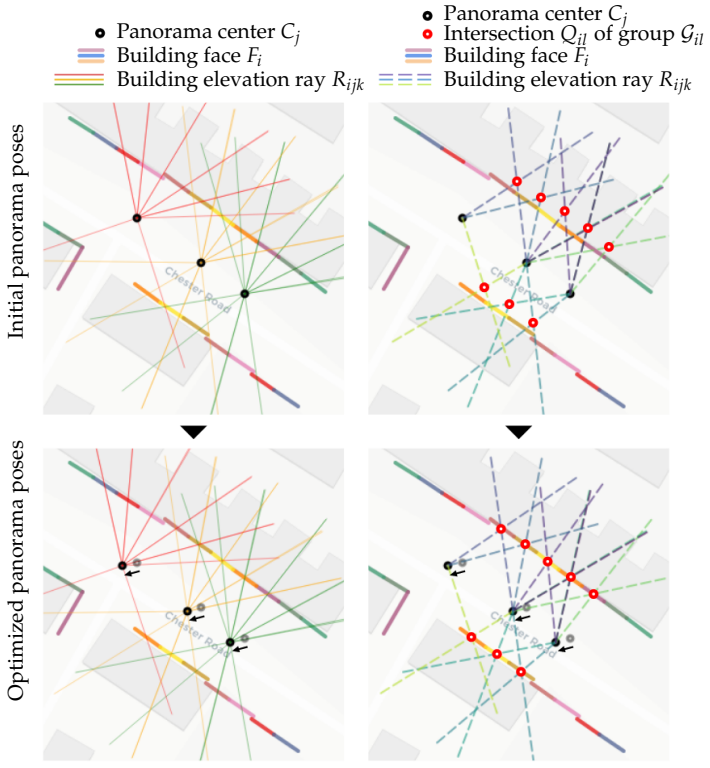

Figure 4. Example of local panorama alignment (top row: initial; bottom row: optimized). On the left, each thin line is a building elevation ray R_{ijk}, i.e., a ray from panorama center C_j through the center of a detected building elevation box E_{ijk} within elevational street view V_{ij}. The color of R_{ijk} matches the panorama P_j it belongs to. After the alignment (dark arrows), each building elevation ray R_{ijk} goes through the center of its intended building face F_i. On the right, each dashed line is a building elevation ray R_{ijk} colored according to the group $\mathcal{G}_{il}$ it belongs to. The intersection Q_{il} of the rays within $\mathcal{G}_{il}$ is marked by a red circle. After the alignment (dark arrows), each intersection Q_{il} lies on the intended building face F_i.

4. Results

As stated in Section 1, this paper focuses on long rows of terraced houses where the lack of space between similar-looking buildings makes visual identification much more sensitive to geolocation errors. Cities such as London, in which swathes of terraced houses were built at a time by developers using variants of standard housing designs [80] are characterized by local similarities but also global variations in their size, appearance, and style. This makes London a good case study to analyze the challenges of visual identification such as misalignment between geospatial data and SVI, and evaluate the method proposed in this paper. A sample of all the annotations (building elevation detection, similarity, and identification), the code, the weights of the trained similarity network, and the identification results are provided as supplementary materials.

4.1. Data Acquisition and Annotation

4.1.1. Data Sources

Two main data sources were used in this study. The first is Google Street View (GSV), which has the best coverage across London. GSV provides a public API which allows users to specify the field of view, heading, and pitch of a virtual camera within a panorama and download the corresponding perspective view. This interface, however, is both inefficient and insufficient for large-scale visual identification: many different elevational street views need to be rendered from each panorama, and valuable metadata (such as the panorama's orientation) are not accessible this way. Therefore, the public API was only used to find panoramas in each area (described below) by uniformly sampling inside the area's georeferenced boundary and querying the panorama closest to each point. Then, a different API described in previous works [81,82] was used to download the whole panorama along with its metadata. The second data source is the Ordnance Survey (https://www.ordnancesurvey.co.uk/, (accessed on 17 February 2024)), which provides a detailed topographic map of building polygons along with various elevation data for each polygon, including the elevation of its highest point.

Buildings were not randomly sampled but selected from nine Lower-Layer Super Output Areas (LSOAs) in London with a high level of repeated property types and built in the same period using data from the Valuation Office Agency (VOA) [83]. Sampling adjacent buildings within each LSOA evaluates the method's ability to distinguish between similar-looking buildings, while selecting LSOAs that span a variety of locations and build periods evaluates the method across different building configurations and styles (see spatial distribution in Figure 5, and temporal and type distribution in Table 1). Their GIS boundaries and dwelling statistics are provided by the Greater London Authority and the VOA, respectively.

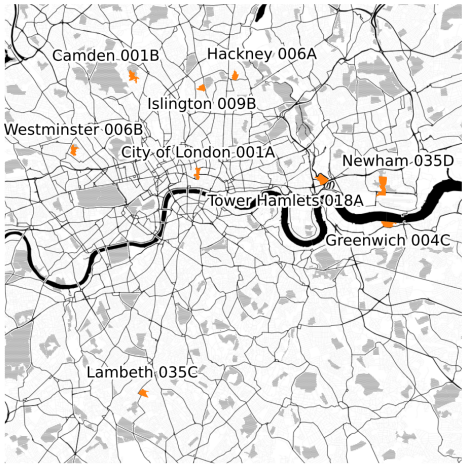

Figure 5. The areas sampled in this work span various parts of London, featuring different architectural styles and build periods. Each area is shown in orange next to its name. [Map tiles by Stamen Design, CC-BY 3.0—Map data ©OpenStreetMap contributors].

Table 1. Property statistics and the number of buildings with at least one elevational street view, broken down by area. Flats/maisonettes include converted terraced houses.

LSOA	Main Build Period	Main Prop. Types (2015)		Buildings with ESV
		Flats/ Maisonettes	Terraced Houses	
Camden 001B	1919–1929	700	40	109
City of London 001A	1965–1972	1090	10	5
Greenwich 004C	1993–1999	430	190	152
Hackney 006A	Pre 1900	510	150	251
Islington 009B	1945–1954	620	50	45
Lambeth 035C	1930–1939	0	450	440
Newham 035D	1983–1992	100	340	379
Tower Hamlets 018A	2000–2015	360	120	137
Westminster 006B	1900–1918	770	0	73

4.1.2. Similarity Annotation

The building elevation similarity dataset is produced by reusing the dataset of annotated building elevation boxes in the following way. Each combination of elevation boxes from different views of the same building face is considered an image pair. Thus, a face F_i with $|\mathcal{V}_i|$ elevational street views, where view V_{ij} contains n_{ij} visible elevations, produces $\sum_{j<j'\leq|\mathcal{V}_i|} n_{ij}n_{ij'}$ pairs. Each pair is then labeled by the annotators as "same", "different", or "unknown". Although the sequentiality of the combinations made the task easier for annotators (who could rely on the memory of recent matches), errors still occurred for particularly similar elevations. These were later automatically detected using a logical rule: positive matches must form a consistent graph, meaning that if A matches B and B matches C, A and C must match. A special case of this rule is that no two elevations from the same view can match the same elevation from another view. The detected annotation errors, which represented 0.8 % of the dataset, were manually corrected.

Comparisons were kept local because the panorama geolocation data are accurate enough to ensure that visual identification only needs to differentiate between neighboring building faces. Nevertheless, the resulting samples also present some interesting properties compared to typical face or vehicle similarity datasets. First, the positive and negative pairs are remarkably balanced, with 48.9% positive pairs. Second, while two adjacent elevations are never completely identical, they also tend to be similar enough that the network has to learn which features are discriminative enough for the task. Compared to randomly pairing images, this makes the level of difficulty more homogeneous across samples.

4.1.3. Identification Annotation

In order to evaluate different identification methods, a final round of annotation was performed to create a ground truth for visual identification. Each bounding box previously drawn in an elevational street view was labeled as "true" when matching the intended building face or otherwise as "false". Manually identifying each bounding box from scratch is time-consuming, especially for long stretches of adjacent buildings, requiring one to go back and forth between the street view image and the map, while keeping in mind the elevations already identified in the street. Therefore, an iterative method was used, whereby the first round of visual identification was performed automatically, followed by a much quicker round of manual corrections. The method used in this round was similar to that described in Section 3, except that the process of similarity estimation was replaced using annotated match labels, and groups were determined by extracting positive-match subgraphs instead of using spectral clustering.

4.1.4. Annotation Statistics

An initial set of 3444 street view panoramas were collected from the nine LSOAs. Of these, 1379 panoramas captured building faces that satisfied the viewing quality criteria

(defined in Section 3.2) and were used to render 5017 elevational street views (4 per panorama on average), representing 1591 buildings (see Table 1 for a breakdown by area). On average, this resulted in 1 visible face and 3 elevational street views per building.

The 2000 annotated elevational street views yielded 2787 bounding boxes, which were combined following the procedure described in Section 4.1.2 to produce 4157 image pairs for similarity estimation, and 1198 of the 2000 views (corresponding to 2046 bounding boxes) were also annotated with an identity following the method in Section 4.1.3. A small proportion of the building elevation boxes (3.1% out of 2046) could not be unambiguously identified because they covered either several building faces (including the intended one) or only a part of the intended face. This map–image misalignment is of a different kind from the one explored in this work, and is further described in Section 4.4.2. The final identification dataset covered 544 building faces (from 453 buildings).

4.2. Quantitative Evaluation

4.2.1. Training

Both the elevation detector and the elevation similarity estimator were implemented in Pytorch and trained on an Ubuntu workstation with an Intel Xeon W-2275 CPU (28 cores at 3.30 GHz), 64 GB of RAM, and an NVIDIA RTX A5000 GPU (24 GB of RAM).

The metric used for the elevation detector was the average precision (AP) [84], which is used to evaluate ranked predictions (i.e., in this case, candidate bounding boxes with a confidence score). The AP at X% (noted AP_X) is a specific case where a candidate bounding box is only considered correct if it overlaps X% of the ground truth bounding box. The elevation detector achieves an AP of 73% on the test set, with an AP_{75} of 84%. For comparison, a recent work by Zhu et al. [85] achieves 51% AP and 57% AP_{75} on a similar façade identification task. Possible explanations for this significant increase include the use of a better backbone network (not specified by Zhu et al.) and the focus on a more architecturally homogeneous dataset (Zhu et al.'s data are from various cities).

The accuracy of the elevation similarity estimator on the test set is 93%. In particular, replacing the backbone's last spatial pooling layer with a channel pooling layer (as described in Section 3.5) resulted in an increase of 11% in accuracy. The F_1 score on the test set is also 93%, due to the elevation similarity dataset being quite balanced between the positive and negative pairs (as noted in Section 4.1.2). On the machine described above, the elevation similarity estimator has an inference speed of 0.023 s per image pair (without batching).

4.2.2. Identification

This quantitative evaluation focuses on the ability of group-aligned matching to overcome the limitations of previous identification approaches when dealing with dense rows of architecturally similar buildings. Only coordinate-based view clipping and instance matching (described in Section 2) are evaluated because feature-based alignment operates under different assumptions (available 3D building model and/or visible building corners). Previous methods are evaluated separately because they produce different results: coordinate-based view clipping only outputs a final image, while instance matching, like group-aligned matching, outputs a set of bounding boxes for each image, assigning a single box per image to the intended building face. To demonstrate the importance of elevation grouping in the identification process, an ablation study is performed by evaluating a method called "instance-aligned matching" that performs panorama alignment without groups (only using the building elevation objective described in Section 3.7).

One way to evaluate the effectiveness of coordinate-based view clipping is to analyze the impact of image margins on the full inclusion of the intended elevation. While elevational street views can be created from the tightest view windows given by the building footprint map, adding a margin on each side of the view increases the chance of fully including the intended elevation. Adding excessive margin, however, also increases the chance of including unintended elevations within the view. This trade-off is clearly shown in Figure 6, where the percentage of elevational street views displaying fully included

elevations is measured as a function of the relative margin added to all sides of each image (in percentage of image width or height, depending on the side). This percentage is broken down into elevational street views that only fully include the intended elevation, only fully include one or more unintended elevation(s), or fully include both. Unsurprisingly, the global percentage increases with the relative margin, but starts to plateau for higher margin values. This is likely due to issues inherent to SVI (especially occlusions and privacy blurring), which affect the elevations' visibility no matter the margin. The trade-off clearly appears when the percentage of elevational street views only fully include the intended elevation peaks at around 40% at 60% margin, before getting mixed with unintended elevations. At the same margin, however, around 10% of elevational street views still only fully include unintended elevations. This demonstrates the limitations of coordinate-based view clipping: minimizing the inclusion of unintended elevations (i.e., a 0% margin) significantly reduces the inclusion of full intended elevations (around 10%). Yet, maximizing the inclusion of full intended elevations necessarily increases the inclusion of full unintended elevations, without any way of differentiating the two.

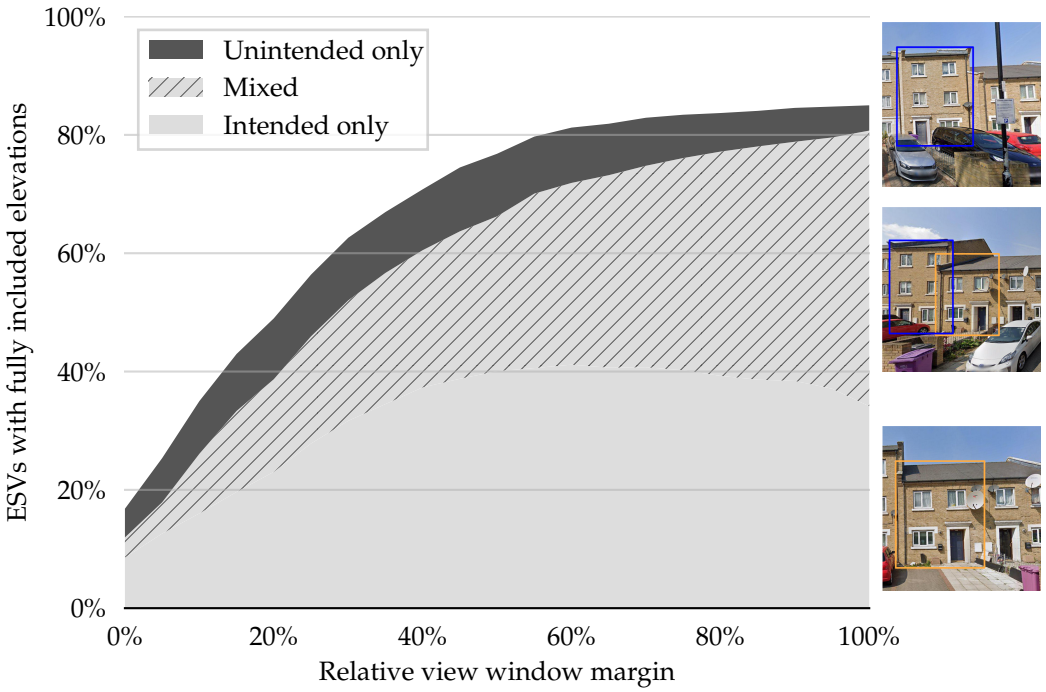

Figure 6. Evaluating coordinate-based view clipping. This graph shows the percentage of elevational street views containing fully included elevations as a function of the relative margin added to expand the view window. The percentage is broken down into the kind of elevations fully included in the views (from top to bottom): unintended elevation(s) only, mix of intended and unintended elevations, and intended elevation only. Corresponding examples of elevational street views are shown on the right, with the intended elevation box in orange and unintended boxes in blue.

Instance, instance-aligned, and group-aligned matching are compared in two scenarios: single- and multi-view. The expected output of the single-view scenario is a single building elevation per face, rendered from the panorama whose origin is closest to the face. Although panorama alignment is still performed using all the elevational street views of each face, only the result for the closest elevational street view is kept when computing the metrics. In the multi-view scenario, instance matching is performed independently for each elevational street view of the same building face, while aligned and group-aligned matching are performed as described. In order to compare methods in an optimal setting and focus on

the building elevation matching part of the pipeline, the same annotated bounding boxes are used for all.

The metrics used to compare methods are accuracy and F_1 score. While accuracy evaluates predictions on both positive and negative building elevations, the F_1 score focuses on the positive ones, which matters more from an information retrieval perspective. It is defined as the harmonic mean of precision and recall, where precision is the ratio of correctly predicted positive labels over predicted positive labels (including erroneous ones), and recall is the ratio of correctly predicted positive labels over positive labels (including those missed by the method). All metrics are computed per building face, meaning that all building elevations in the views of a given face have to be correctly labeled for it to be counted as correctly identified. In the single-view scenario, building elevations only come from one elevational street view per face, but in the multi-view scenario, predicted labels are aggregated over all the elevational street views of a face. As shown in Table 2, group-aligned matching outperforms instance matching for both metrics, particularly in the more challenging multi-view scenario. Instance-aligned matching performs worse than group-aligned matching, and even worse than instance matching in the single-view scenario, showing that simply pushing building elevation boxes towards their closest building face can reinforce initial misalignments instead of correcting them. This demonstrates that finding building elevation groups across panoramas is an essential part of the pipeline.

Table 2. Evaluating bounding box-based methods in the single- and multi-view scenarios. Instance matching is used by the previous works described in Section 2.3. Instance-aligned matching is an ablated version of group-aligned matching. The best results for each metric are shown in bold.

	Single-View		Multi-View	
Method	**Accuracy**	**F_1**	**Accuracy**	**F_1**
Instance matching	90%	92%	79%	85%
Instance-aligned matching	88%	92%	83%	90%
Group-aligned matching	**93%**	**96%**	**88%**	**93%**

4.3. Qualitative Results

4.3.1. Similarity Estimation

Figure 7 shows examples of building elevation pairs whose similarity is correctly estimated. The left column contains two easier examples for which the network outputs highly confident similarity scores (very low for different building faces and very high for the same building face). The negative pair is made easier by the large change in viewing angle and visual context around building elevations. Conversely, in the positive case, the viewing angles and contexts are very similar. The right column contains two of the pairs that the network is the least confident about (giving a similarity score close to 0.5). In both cases, the building face is only partially visible due to occlusions, and for the positive pair, the bottom part changes significantly.

Failure cases are shown in Figure 8. Here, the similarity estimator is very confidently wrong. The positive pair with a low similarity score features significant changes in context due to the very different panorama locations. The negative pair with a high similarity score comes from a street where buildings are particularly similar and clues that could help the estimation (such as the position of the street lamp) are very thin.

4.3.2. Identification

Figure 9 shows two examples of visual identification. Due to the misalignment between panorama pose data and building footprints, most building elevation rays fail to go through the right building face. In the first example, instance matching erroneously matches the only ray crossing the intended face in (a), and does not match anything in (b). The building elevation objective is enough to correctly align the rays since both the aligned and group-aligned methods correctly match the building elevations. This objective includes

all the building elevation rays originating from panoramas (a–c), including those going through neighboring building faces (omitted in the map), which helps find panorama poses more consistent with the surrounding building faces. In the second example, instance matching fails to find the right building elevation box in (a) because both rays go around the intended face without intersecting with it. Instance-aligned matching fails in (b), potentially due to the building elevation objective shifting the panorama in such a way that none of the rays intersect the intended face. Adding the building elevation group objective, however, ensures that building elevations that were grouped together by the similarity network keep consistent ray directions during the optimization, resulting in all three rays intersecting with the intended face.

Figure 7. Successful examples of pairwise similarity estimation (**top** row: different building faces, orange frame; **bottom** row: same building faces, green frame). The score indicates the network's confidence (from 0 to 1) that the central elevations come from the same building face.

Figure 8. Examples of failure cases of pairwise similarity estimation (**top** row: same building faces, green frame; **bottom** row: different building faces, orange frame).

Figure 9. Examples of identification results. The left panel is a map showing the intended (thick orange) and unintended (thick blue) building faces, three panoramas close to the intended building face, and the building elevation rays going through the elevation boxes' centers (marked by a cross) detected in each panorama. Each panorama provides a single elevational street view for the intended building face, shown on the right. Each elevational street view contains detected building elevation boxes with the same color scheme as the map. The colored dots under each box show the box selected by each identification method. For example, the (blue) unintended elevation box in (**a**) corresponds to the (thin blue) unintended elevation ray in the map that appears to go from the center of the panorama (**a**) through the intended building face, which is why this elevation box is erroneously selected by the instance matching method (shown by the pink dot underneath the box). However, since panoramas and building faces are misaligned, this ray does not actually go through the intended building face. Images (**b**–**f**) have a similar correspondence between building elevation rays in the map and building elevation boxes in the elevational street view.

4.4. Discussion

4.4.1. Results

While the results in Sections 4.2.2 and 4.3.2 validate the effectiveness of group-aligned matching, comparing its accuracy to other classes of building identification methods is challenging because of the different inputs, outputs, and assumptions inherent to each. Coordinate-based view clipping, which only uses geometric parameters, does not produce a bounding box, but its accuracy was approximated by taking a bounding box-annotated dataset and checking how often clipped views contain their intended box without containing an adjacent box. As shown in Figure 6, this accuracy peaks slightly above 40%. However, both this value and the corresponding value of the window margin parameter (around 60%) most likely depend on the specific characteristics of the evaluation dataset (including the average spacing between buildings and the average magnitude of panorama pose errors). Meanwhile, feature-based alignment was not evaluated due to the lack of a

georeferenced 3D model and the difficulty of reliably detecting the corners and vertical edges of terraced houses, which make up a large portion of the evaluation dataset. It is worth noting that that a recent feature-based alignment approach [44] achieved a (single-view) building identification accuracy of 83.5%, although the difference between evaluation datasets makes this result difficult to compare to the accuracy reported in the present work.

Instance matching, which forms the basis upon which group-aligned matching builds, is more straightforward to compare. As shown in Table 2, the accuracy of group-aligned matching in the single-view scenario is 93%, a gain of 3% compared to instance matching. Similarly, the F_1 score of group-aligned matching is 96%, representing a gain of 4% compared to instance matching. To put these results in perspective, the two most recent works using instance matching report an identification accuracy of 90.5% [47] and 99.6% [46], respectively, and for the latter work, an F_1 score of 87.9% [46], although once again, the difference between the urban settings considered in each work precludes accurate comparisons. In the multi-view scenario, the accuracy of group-aligned matching is 88%, outperforming instance matching by 9%, and the F_1 score of group-aligned matching is 93 %, outperforming instance matching by 8%. These results support the argument that retrieving multiple views per building face allows visual consistency checks that increase the robustness of identification. Moreover, the ability to view building faces from different angles increases the visibility of the features of interest, which increases their usefulness for further visual analysis in downstream applications.

The accuracy gain of group-aligned matching compared to instance matching is 3% in the single-view scenario and 9% in the multi-view scenario. This difference can be partially explained by a geometric argument. The farther a panorama origin C_j is from a building face F_i, and the larger the angle α_n (defined in Section 3.2), the more panorama pose errors impact the geometric parameters of the elevational street view V_{ij}. Since farther panoramas are used in the multi-view scenario, their pose errors have a stronger impact on the corresponding elevational street views. The views from these additional panoramas are more likely to be shifted in a way that compromises the accuracy of instance matching.

These findings have both theoretical and practical implications for the wider field of urban analytics. With regards to theory, previous works have argued for the epistemic advantages of street view imagery (SVI) by providing a unique point of view and a greater level of detail compared to other, more commonly used sources of imagery [3]. Our findings nuance this argument by showing that SVI pose errors and occlusions can seriously affect the reliability of building-level analysis, especially in the case of narrow, densely packed buildings. A related theoretical question is whether retrieving multiple views per building face consistently improves the accuracy of downstream analysis across a wide variety of tasks. While answering this question is beyond the scope of this paper, our findings suggest that performing multi-view analysis requires paying particular attention to the way that multiple views are collected and assigned to the same building face. In practical terms, the first implication of our work is that methods that rely on SVI to perform building-level analysis should include an assessment of the rate of misidentified buildings in their data. Second, collecting multiple views per building face is not only useful for downstream applications (notably to work around occlusions), but can also be leveraged at the identification stage to ensure that views are consistent, thus increasing the robustness of the linkage between georeferenced models and SVI.

4.4.2. Limitations and Future Work

As shown in Sections 4.2.1 and 4.3.1, both the elevation detector and elevation similarity estimator perform well on their respective test sets. Using deep convolutional networks for both tasks gives access to high-level visual features that are more useful than corners and edges to separate and differentiate buildings faces that are adjacent and architecturally similar. While these results are encouraging, the models should be tested in different cities around the world with similar urban layouts to evaluate their generalization performance and quantify the "domain gap" [62]. While more diverse data could improve generalization,

training location-specific models instead could provide higher accuracy. More research is needed to quantitatively evaluate this trade-off.

Group-aligned matching has two main limitations, as illustrated in Figure 10. First, panorama pose misalignment can occasionally be difficult to correct due to a lack of neighboring building faces (for example, if buildings are only on one side of the street). Even if building elevation rays are grouped together, all the groups may simultaneously shift so that all building elevations are off by one building face. While street segments featuring buildings on a single side are relatively uncommon in the evaluation dataset, an additional geometric analysis could be performed during the first step (Section 3.2) to detect building configurations that may not provide enough information to realign the panoramas accurately.

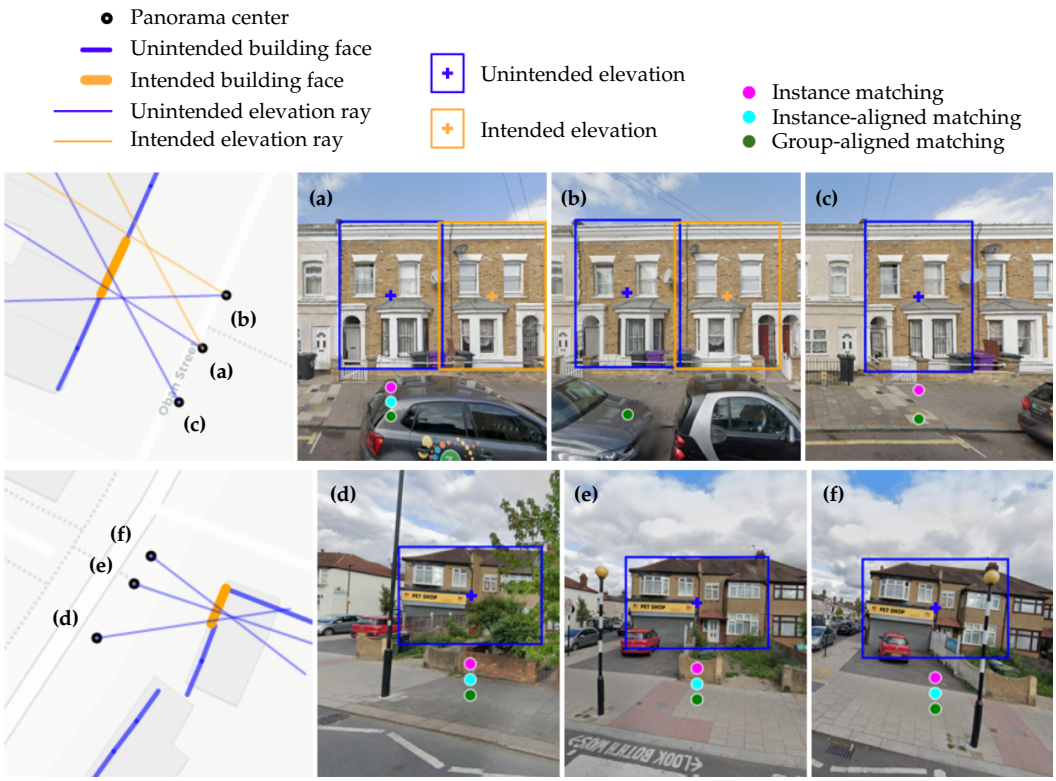

Figure 10. Some failure cases of group-aligned matching. As in Figure 9, each panorama provides a single elevational street view (**a**–**f**) for the intended building face, shown as a thick orange line segment. For each panorama, the colored building elevation rays in the map on the left match the colored elevation boxes in the corresponding image on the right. Legend is the same as Figure 9.

The second limitation happens when (geometric) building faces cannot be properly matched to (visual) building elevations, or vice versa, for example, because the data sources are not synchronized (a building may have been constructed, demolished, subdivided, or extended) or because the street-level perspective does not provide enough context to unambiguously separate building elevations. The second example in Figure 10 shows two building faces erroneously annotated as one because they are visually homogeneous and the left face does not seem to have a door (which is actually on the side of the building). This kind of misalignment between footprints and SVI is topological rather than spatial, and is not specific to group-aligned matching. It is worth noting, however, that only a small proportion of the building elevation bounding boxes (3.1%) were found to have this issue in the evaluation dataset (as detailed in Section 4.1.4). Integrating additional visual data from more frequently updated sources (such as satellite imagery), following an approach similar to that of Cao et al. [86], is an interesting direction to address this limitation.

In terms of methodological limitations, the pipeline presented in this paper involves various parameters and assumptions whose impact on the overall performance remains to

be evaluated, especially around the first step (finding exposed building faces, Section 3.2) and the last (panorama alignment, Section 3.7). While the effect of image margins was quantified in Section 4.2.2, and deep learning hyperparameters for each model were tuned on their respective validation sets, more analysis is required to better understand the sensitivity of the method.

5. Conclusions

This paper introduces a new visual identification method that generates a multi-view database linking building footprints with views extracted from SVI. Focusing on the dense rows of narrow houses with a high level of repetition raises a challenge that previous identification methods are not equipped to tackle due to panorama misalignment, occlusions, and the lack of geometric features separating building faces. Group-aligned matching, in contrast, combines the advantages of previous approaches by (i) locating individual building elevations using a deep neural network and (ii) performing model–image alignment using these detected elevations. In addition, it increases the robustness of visual identification by collecting multiple views per building face and ensuring their visual consistency using a deep Siamese network. This yields significant gains in accuracy and F_1 score, particularly in the multi-view scenario. The main conclusion of this work is that previous identification methods lack robustness to panorama pose errors when buildings are narrow, densely packed, and subject to occlusions, while collecting multiple views per building can be leveraged to increase the robustness of visual identification by ensuring that building views are consistent.

Multi-view identification opens many avenues for future research. From a computer vision perspective, creating a multi-view dataset enables applications such as building re-identification for geo-localization, view synthesis, and 3D reconstruction. Reliably indexing building views at scale also provides an essential linkage between visual and statistical data sources, enabling multi-modal building-level analysis with many applications in architecture and urban studies, including the fine-grained mapping of building typology, style, level of upkeep, and real estate value. The analysis of transformations over time is another promising research area that comes with its own challenges due to the lack of synchronization between maps and SVI [6], linking to the problem of "persistent place recognition" explored in robotics [87]. Lastly, the potential privacy issues raised by the cross-referencing of data sources at the building level [3] need to be assessed and addressed, for example, taking inspiration from research in person re-identification [88].

Supplementary Materials: The following supporting information can be downloaded at: https://www.mdpi.com/article/10.3390/buildings14030578/s1.

Author Contributions: Conceptualization, R.R., S.J. and A.A.; methodology, R.R.; software, R.R.; validation, R.R.; formal analysis, R.R.; investigation, R.R.; resources, S.J. and A.A.; data curation, R.R.; writing—original draft preparation, R.R.; writing—review and editing, S.J. and A.A.; visualization, R.R.; supervision, S.J. and A.A.; project administration, S.J.; funding acquisition, S.J. and A.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Prosit Philosophiae Foundation.

Data Availability Statement: The images, annotations, and geospatial data used in this study are available upon request from the corresponding author. The data are not publicly available due to copyright restrictions.

Conflicts of Interest: The authors declare no conflicts of interest. The funder had no role in the design, execution, interpretation, or writing of this study.

References

1. Rzotkiewicz, A.; Pearson, A.L.; Dougherty, B.V.; Shortridge, A.; Wilson, N. Systematic review of the use of Google Street View in health research: Major themes, strengths, weaknesses and possibilities for future research. *Health Place* **2018**, *52*, 240–246. [CrossRef]
2. He, N.; Li, G. Urban neighbourhood environment assessment based on street view image processing: A review of research trends. *Environ. Chall.* **2021**, *4*, 100090. [CrossRef]
3. Cinnamon, J.; Jahiu, L. Panoramic Street-Level Imagery in Data-Driven Urban Research: A Comprehensive Global Review of Applications, Techniques, and Practical Considerations. *Isprs Int. J. Geo-Inf.* **2021**, *10*, 471. [CrossRef]
4. Li, Y.; Peng, L.; Wu, C.; Zhang, J. Street View Imagery (SVI) in the Built Environment: A Theoretical and Systematic Review. *Buildings* **2022**, *12*, 1167. [CrossRef]
5. Starzyńska-Grześ, M.B.; Roussel, R.; Jacoby, S.; Asadipour, A. Computer vision-based analysis of buildings and built environments: A systematic review of current approaches. *Acm Comput. Surv.* **2023**, *55*, 1–25. [CrossRef]
6. Biljecki, F.; Ito, K. Street view imagery in urban analytics and GIS: A review. *Landsc. Urban Plan.* **2021**, *215*, 104217. [CrossRef]
7. Liu, L.; Silva, E.A.; Wu, C.; Wang, H. A Machine Learning-Based Method for the Large-Scale Evaluation of the Qualities of the Urban Environment. *Comput. Environ. Urban Syst.* **2017**, *65*, 113–125. [CrossRef]
8. Shen, Q.; Zeng, W.; Ye, Y.; Arisona, S.M.; Schubiger, S.; Burkhard, R.; Qu, H. StreetVizor: Visual Exploration of Human-Scale Urban Forms Based on Street Views. *IEEE Trans. Vis. Comput. Graph.* **2018**, *24*, 1004–1013. [CrossRef] [PubMed]
9. Li, Y.; Chen, Y.; Rajabifard, A.; Khoshelham, K.; Aleksandrov, M. Estimating Building Age from Google Street View Images Using Deep Learning. In Proceedings of the 10th International Conference on Geographic Information Science (GIScience), Melbourne, Australia, 28–31 August 2018, Volume 114; pp. 40:1–40:7. [CrossRef]
10. Lindenthal, T.; Johnson, E.B. Machine Learning, Architectural Styles and Property Values. *J. Real Estate Financ. Econ.* **2021** . [CrossRef]
11. Varghese, A.; Gubbi, J.; Ramaswamy, A.; Balamuralidhar, P. ChangeNet: A Deep Learning Architecture for Visual Change Detection. In Proceedings of the European Conference on Computer Vision (ECCV) Workshops, Munich, Germany, 8–14 September 2018 ; pp. 129–145. [CrossRef]
12. Sakurada, K.; Shibuya, M.; Wang, W. Weakly Supervised Silhouette-based Semantic Change Detection. In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 31 May–31 August 2020; IEEE: Piscataway, NJ, USA , 2020; pp. 6861–6867. [CrossRef]
13. Dai, M.; Ward, W.O.; Meyers, G.; Densley Tingley, D.; Mayfield, M. Residential building facade segmentation in the urban environment. *Build. Environ.* **2021**, *199*, 107921. [CrossRef]
14. Fond, A.; Berger, M.O.; Simon, G. Model-image registration of a building's facade based on dense semantic segmentation. *Comput. Vis. Image Underst.* **2021**, *206*, 103185. [CrossRef]
15. Hu, H.; Wang, L.; Zhang, M.; Ding, Y.; Zhu, Q. Fast and Regularized Reconstruction of Building Façades from Street-View Images using Binary Integer Programming. *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.* **2020**, *V-2-2020*, 365–371. [CrossRef]
16. Weyand, T.; Araujo, A.; Cao, B.; Sim, J. Google Landmarks Dataset v2—A Large-Scale Benchmark for Instance-Level Recognition and Retrieval. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 2572–2581. [CrossRef]
17. Krylov, V.A.; Kenny, E.; Dahyot, R. Automatic Discovery and Geotagging of Objects from Street View Imagery. *Remote Sens.* **2018**, *10*, 661. [CrossRef]
18. Campbell, A.; Both, A.; Sun, Q.C. Detecting and mapping traffic signs from Google Street View images using deep learning and GIS. *Comput. Environ. Urban Syst.* **2019**, *77*, 101350. [CrossRef]
19. Laumer, D.; Lang, N.; Van Doorn, N.; Mac Aodha, O.; Perona, P.; Wegner, J.D. Geocoding of trees from street addresses and street-level images. *Isprs J. Photogramm. Remote Sens.* **2020**, *162*, 125–136. [CrossRef]
20. Liu, D.; Jiang, Y.; Wang, R.; Lu, Y. Establishing a citywide street tree inventory with street view images and computer vision techniques. *Comput. Environ. Urban Syst.* **2023**, *100*, 101924. [CrossRef]
21. Pylvänäinen, T.; Roimela, K.; Vedantham, R.; Wang, R.; Grzeszczuk, R. Automatic Alignment and Multi-View Segmentation of Street View Data using 3D Shape Priors. In Proceedings of the 5th International Symposium on 3D Data Processing, Visualization and Transmission (3DPVT), Paris, France, 17–20 May 2010 .
22. Kelly, T.; Femiani, J.; Wonka, P.; Mitra, N.J. BigSUR: Large-scale Structured Urban Reconstruction. *ACM Trans. Graph.* **2017**, *36*, 1–16. [CrossRef]
23. Thackway, W.; Ng, M.; Lee, C.L.; Pettit, C. Implementing a deep-learning model using Google street view to combine social and physical indicators of gentrification. *Comput. Environ. Urban Syst.* **2023**, *102*, 101970. [CrossRef]
24. Zou, S.; Wang, L. Detecting individual abandoned houses from google street view: A hierarchical deep learning approach. *Isprs J. Photogramm. Remote Sens.* **2021**, *175*, 298–310. [CrossRef]
25. Law, S.; Paige, B.; Russell, C. Take a Look Around: Using Street View and Satellite Images to Estimate House Prices. *ACM Trans. Intell. Syst. Technol.* **2019**, *10*, 1–19. [CrossRef]
26. Wang, C.; Antos, S.E.; Triveno, L.M. Automatic detection of unreinforced masonry buildings from street view images using deep learning-based image segmentation. *Autom. Constr.* **2021**, *132*, 103968. [CrossRef]

27. Yang, F.; Wang, M. Deep Learning-Based Method for Detection of External Air Conditioner Units from Street View Images. *Remote Sens.* **2021**, *13*, 3691. [CrossRef]
28. Wang, C.; Antos, S.E.; Gosling-Goldsmith, J.G.; Triveno, L.M.; Zhu, C.; von Meding, J.; Ye, X. Assessing Climate Disaster Vulnerability in Peru and Colombia Using Street View Imagery: A Pilot Study. *Buildings* **2024**, *14*, 14. [CrossRef]
29. Mai, W.; Tweed, C.; Hung, P. Building Identification by Low-Resolution Mobile Images. In Proceedings of the 5th International Conference on Advances in Mobile Computing and Multimedia (MoMM), Jakarta, Indonesia, 3–5 December 2007; Volume 230, pp. 15–30.
30. Cham, T.J.; Ciptadi, A.; Tan, W.C.; Pham, M.T.; Chia, L.T. Estimating camera pose from a single urban ground-view omnidirectional image and a 2D building outline map. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), San Francisco, CA, USA, 13–18 June 2010; IEEE: Piscataway, NJ, USA, 2010; pp. 366–373. [CrossRef]
31. Chen, D.M.; Baatz, G.; Köser, K.; Tsai, S.S.; Vedantham, R.; Pylvänäinen, T.; Roimela, K.; Chen, X.; Bach, J.; Pollefeys, M.; et al. City-scale landmark identification on mobile devices. In Proceedings of the CVPR 2011, Colorado Springs, CO, USA, 20–25 June 2011; pp. 737–744. [CrossRef]
32. Torii, A.; Sivic, J.; Pajdla, T.; Okutomi, M. Visual Place Recognition with Repetitive Structures. In Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 883–890. [CrossRef]
33. Chu, H.; Gallagher, A.; Chen, T. GPS Refinement and Camera Orientation Estimation from a Single Image and a 2D Map. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Columbus, OH, USA, 23–28 June 2014; IEEE: Piscataway, NJ, USA, 2014; pp. 171–178. [CrossRef]
34. Lee, K.; Lee, S.; Jung, W.J.; Kim, K.T. Fast and Accurate Visual Place Recognition Using Street-View Images. *ETRI J.* **2017**, *39*, 97–107. [CrossRef]
35. Karlekar, J.; Zhou, S.Z.; Lu, W.; Loh, Z.C.; Nakayama, Y.; Hii, D. Positioning, tracking and mapping for outdoor augmentation. In Proceedings of the 2010 IEEE International Symposium on Mixed and Augmented Reality, Nantes, France, 9–13 October 2010; pp. 175–184. [CrossRef]
36. Arth, C.; Pirchheim, C.; Ventura, J.; Schmalstieg, D.; Lepetit, V. Instant Outdoor Localization and SLAM Initialization from 2.5D Maps. *IEEE Trans. Vis. Comput. Graph.* **2015**, *21*, 1309–1318. [CrossRef] [PubMed]
37. Fond, A.; Berger, M.O.; Simon, G. Facade Proposals for Urban Augmented Reality. In Proceedings of the IEEE International Symposium on Mixed and Augmented Reality (ISMAR), Nantes, France, 9–13 October 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 32–41. [CrossRef]
38. Xiao, J.; Fang, T.; Zhao, P.; Lhuillier, M.; Quan, L. Image-based street-side city modeling. *ACM Trans. Graph.* **2009**, *28*, 1–12. [CrossRef]
39. Chu, H.; Wang, S.; Urtasun, R.; Fidler, S. HouseCraft: Building Houses from Rental Ads and Street Views. In Proceedings of the Computer Vision–ECCV 2016, Amsterdam, The Netherlands, 11–14 October 2016; Leibe, B., Matas, J., Sebe, N., Welling, M., Eds.; Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2016; pp. 500–516. [CrossRef]
40. Liu, P.; Biljecki, F. A review of spatially-explicit GeoAI applications in Urban Geography. *Int. J. Appl. Earth Obs. Geoinf.* **2022**, *112*, 102936. [CrossRef]
41. Kang, J.; Körner, M.; Wang, Y.; Taubenböck, H.; Zhu, X.X. Building Instance Classification Using Street View Images. *Isprs J. Photogramm. Remote Sens.* **2018**, *145*, 44–59. [CrossRef]
42. Mayer, K.; Haas, L.; Huang, T.; Bernabé-Moreno, J.; Rajagopal, R.; Fischer, M. Estimating building energy efficiency from street view imagery, aerial imagery, and land surface temperature data. *Appl. Energy* **2023**, *333*, 120542. [CrossRef]
43. Xu, Z.; Zhang, F.; Wu, Y.; Yang, Y.; Wu, Y. Building height calculation for an urban area based on street view images and deep learning. *Comput.-Aided Civ. Infrastruct. Eng.* **2023**, *38*, 892–906. [CrossRef]
44. Ogawa, M.; Aizawa, K. Identification of Buildings in Street Images Using Map Information. In Proceedings of the IEEE International Conference on Image Processing (ICIP), Taipei, Taiwan, 22–25 September 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 984–988. [CrossRef]
45. Zhang, C.; Yankov, D.; Wu, C.T.; Shapiro, S.; Hong, J.; Wu, W. What is That Building? An End-to-End System for Building Recognition from Streetside Images. In Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Virtual Event, CA, USA, 6–10 July 2020; pp. 2425–2433. [CrossRef]
46. Khan, S.; Salvaggio, C. Automatically Gather Address Specific Dwelling Images Using Google Street View. In Proceedings of the 25th International Conference on Pattern Recognition (ICPR), Milan, Italy, 10–15 January 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 473–480. [CrossRef]
47. Ogawa, Y.; Oki, T.; Chen, S.; Sekimoto, Y. Joining Street-View Images and Building Footprint GIS Data. In Proceedings of the 1st ACM SIGSPATIAL International Workshop on Searching and Mining Large Collections of Geospatial Data, Beijing, China, 2 November 2021; pp. 18–24. [CrossRef]
48. Li, X.; Zhang, C.; Li, W. Building block level urban land-use information retrieval based on Google Street View images. *GIScience Remote Sens.* **2017**, *54*, 819–835. [CrossRef]
49. Zhang, W.; Li, W.; Zhang, C.; Hanink, D.M.; Li, X.; Wang, W. Parcel-based urban land use classification in megacity using airborne LiDAR, high resolution orthoimagery, and Google Street View. *Comput. Environ. Urban Syst.* **2017**, *64*, 215–228. [CrossRef]
50. Ilic, L.; Sawada, M.; Zarzelli, A. Deep Mapping Gentrification in a Large Canadian City Using Deep Learning and Google Street View. *PLoS ONE* **2019**, *14*, e0212814. [CrossRef]

51. Srivastava, S.; Vargas-Muñoz, J.E.; Tuia, D. Understanding urban landuse from the above and ground perspectives: A deep learning, multimodal solution. *Remote Sens. Environ.* **2019**, *228*, 129–143. [CrossRef]
52. Sharifi Noorian, S.; Qiu, S.; Psyllidis, A.; Bozzon, A.; Houben, G.J. Detecting, Classifying, and Mapping Retail Storefronts Using Street-Level Imagery. In Proceedings of the International Conference on Multimedia Retrieval, Dublin, Ireland, 8–11 June 2020; pp. 495–501. [CrossRef]
53. Yao, Y.; Zhang, J.; Qian, C.; Wang, Y.; Ren, S.; Yuan, Z.; Guan, Q. Delineating urban job-housing patterns at a parcel scale with street view imagery. *Int. J. Geogr. Inf. Sci.* **2021**, *35*, 1927–1950. [CrossRef]
54. Szcześniak, J.T.; Ang, Y.Q.; Letellier-Duchesne, S.; Reinhart, C.F. A Method for Using Street View Imagery to Auto-Extract Window-to-Wall Ratios and Its Relevance for Urban-Level Daylighting and Energy Simulations. *Build. Environ.* **2022**, *207*, 108108. [CrossRef]
55. Taneja, A.; Ballan, L.; Pollefeys, M. Geometric Change Detection in Urban Environments Using Images. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 2193–2206. [CrossRef] [PubMed]
56. Yuan, J.; Cheriyadat, A.M. Combining Maps and Street Level Images for Building Height and Facade Estimation. In Proceedings of the 2nd ACM SIGSPATIAL Workshop on Smart Cities and Urban Analytics, Burlingame, CA, USA, 31 October 2016; pp. 8:1–8:8. [CrossRef]
57. Park, J.; Jeon, I.B.; Yoon, S.E.; Woo, W. Instant Panoramic Texture Mapping with Semantic Object Matching for Large-Scale Urban Scene Reproduction. *IEEE Trans. Vis. Comput. Graph.* **2021**, *27*, 2746–2756. [CrossRef] [PubMed]
58. Ogawa, Y.; Zhao, C.; Oki, T.; Chen, S.; Sekimoto, Y. Deep Learning Approach for Classifying the Built Year and Structure of Individual Buildings by Automatically Linking Street View Images and GIS Building Data. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2023**, *16*, 1740–1755. [CrossRef]
59. Pang, H.E.; Biljecki, F. 3D building reconstruction from single street view images using deep learning. *Int. J. Appl. Earth Obs. Geoinf.* **2022**, *112*, 102859. [CrossRef]
60. Xu, X.; Qiu, W.; Li, W.; Liu, X.; Zhang, Z.; Li, X.; Luo, D. Associations between Street-View Perceptions and Housing Prices: Subjective vs. Objective Measures Using Computer Vision and Machine Learning Techniques. *Remote Sens.* **2022**, *14*, 891. [CrossRef]
61. Chen, C.W.; Kuo, Y.H.; Lee, T.; Lee, C.H.; Hsu, W. Drone-View Building Identification by Cross-View Visual Learning and Relative Spatial Estimation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Salt Lake City, UT, USA, 18–22 June 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 1558–15588. [CrossRef]
62. Ye, M.; Shen, J.; Lin, G.; Xiang, T.; Shao, L.; Hoi, S.C. Deep Learning for Person Re-identification: A Survey and Outlook. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *44*, 2872–2893. [CrossRef] [PubMed]
63. Wang, H.; Hou, J.; Chen, N. A Survey of Vehicle Re-Identification Based on Deep Learning. *IEEE Access* **2019**, *7*, 172443–172469. [CrossRef]
64. Taneja, A.; Ballan, L.; Pollefeys, M. Registration of Spherical Panoramic Images with Cadastral 3D Models. In Proceedings of the 2nd International Conference on 3D Imaging, Modeling, Processing, Visualization & Transmission (3DIMPVT), Zurich, Switzerland, 13–15 October 2012; IEEE: Piscataway, NJ, USA, 2012; pp. 479–486. [CrossRef]
65. Chen, J.; Yu, Z.; Yang, C.; Yang, K. Automatic registration of urban high-resolution remote sensing images based on characteristic spatial objects. *Sci. Rep.* **2022**, *12*, 14432. [CrossRef]
66. Suri, S.; O' Rourke, J. Worst-Case Optimal Algorithms for Constructing Visibility Polygons with Holes. In Proceedings of the 2nd Annual Symposium on Computational Geometry, Yorktown Heights, NY, USA, 2–4 June 1986; pp. 14–23. [CrossRef]
67. Liang, J.; Gong, J.; Sun, J.; Zhou, J.; Li, W.; Li, Y.; Liu, J.; Shen, S. Automatic Sky View Factor Estimation from Street View Photographs—A Big Data Approach. *Remote Sens.* **2017**, *9*, 411. [CrossRef]
68. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In Proceedings of the 28th International Conference on Neural Information Processing Systems, Montreal, QC, Canada, 7–12 December 2015; pp. 91–99.
69. Lin, T.Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature Pyramid Networks for Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 936–944. [CrossRef]
70. Yan, K.; Wang, Y.; Liang, D.; Huang, T.; Tian, Y. CNN vs. SIFT for Image Retrieval: Alternative or Complementary? In Proceedings of the Proceedings of the 24th ACM international conference on Multimedia, Amsterdam, The Netherlands, 15–19 October 2016; MM'16; pp. 407–411. [CrossRef]
71. Ma, J.; Jiang, X.; Fan, A.; Jiang, J.; Yan, J. Image Matching from Handcrafted to Deep Features: A Survey. *Int. J. Comput. Vis.* **2021**, *129*, 23–79. [CrossRef]
72. Lowe, D.G. Distinctive Image Features from Scale-Invariant Keypoints. *Int. J. Comput. Vis.* **2004**, *60*, 91–110. [CrossRef]
73. Bay, H.; Tuytelaars, T.; Van Gool, L. SURF: Speeded Up Robust Features. In Proceedings of the Computer Vision–ECCV 2006, Graz, Austria, 7–13 May 2006; Leonardis, A., Bischof, H., Pinz, A., Eds.; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2006; pp. 404–417. [CrossRef]
74. Zhai, A.; Wu, H.Y. Classification is a Strong Baseline for Deep Metric Learning. In Proceedings of the 30th British Machine Vision Conference (BMVC), Cardiff, UK, 9–12 September 2019; pp. 1–12.

75. Boudiaf, M.; Rony, J.; Ziko, I.M.; Granger, E.; Pedersoli, M.; Piantanida, P.; Ben Ayed, I. A Unifying Mutual Information View of Metric Learning: Cross-Entropy vs. Pairwise Losses. In Proceedings of the European Conference on Computer Vision (ECCV), Glasgow, UK, 23–28 August 2020; pp. 548–564. [CrossRef]
76. von Luxburg, U. A tutorial on spectral clustering. *Stat. Comput.* **2007**, *17*, 395–416. [CrossRef]
77. Rousseeuw, P.J. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *J. Comput. Appl. Math.* **1987**, *20*, 53–65. [CrossRef]
78. Wales, D.J.; Doye, J.P.K. Global Optimization by Basin-Hopping and the Lowest Energy Structures of Lennard-Jones Clusters Containing up to 110 Atoms. *J. Phys. Chem. A* **1997**, *101*, 5111–5116. [CrossRef]
79. Zhu, C.; Byrd, R.H.; Lu, P.; Nocedal, J. Algorithm 778: L-BFGS-B: Fortran Subroutines for Large-Scale Bound-Constrained Optimization. *ACM Trans. Math. Softw.* **1997**, *23*, 550–560. [CrossRef]
80. Muthesius, S. *The English Terraced House*; Yale University Press: New Haven, CT, USA, 1982.
81. Gong, F.Y.; Zeng, Z.C.; Zhang, F.; Li, X.; Ng, E.; Norford, L.K. Mapping Sky, Tree, and Building View Factors of Street Canyons in a High-Density Urban Environment. *Build. Environ.* **2018**, *134*, 155–167. [CrossRef]
82. Li, X.; Ratti, C. Mapping the Spatio-Temporal Distribution of Solar Radiation within Street Canyons of Boston Using Google Street View Panoramas and Building Height Model. *Landsc. Urban Plan.* **2019**, *191*, 103387. [CrossRef]
83. Özer, S.; Jacoby, S. Dwelling size and usability in London: a study of floor plan data using machine learning. *Build. Res. Inf.* **2022**, *50*, 694–708. [CrossRef]
84. Everingham, M.; Van Gool, L.; Williams, C.K.I.; Winn, J.; Zisserman, A. The Pascal Visual Object Classes (VOC) Challenge. *Int. J. Comput. Vis.* **2010**, *88*, 303–338. [CrossRef]
85. Zhu, P.; Para, W.R.; Frühstück, A.; Femiani, J.; Wonka, P. Large-Scale Architectural Asset Extraction from Panoramic Imagery. *IEEE Trans. Vis. Comput. Graph.* **2020**, *28*, 1301–1316. [CrossRef]
86. Cao, R.; Zhu, J.; Tu, W.; Li, Q.; Cao, J.; Liu, B.; Zhang, Q.; Qiu, G. Integrating Aerial and Street View Images for Urban Land Use Classification. *Remote Sens.* **2018**, *10*, 1553. [CrossRef]
87. Lowry, S.; Sünderhauf, N.; Newman, P.; Leonard, J.J.; Cox, D.; Corke, P.; Milford, M.J. Visual Place Recognition: A Survey. *IEEE Trans. Robot.* **2016**, *32*, 1–19. [CrossRef]
88. Yaghoubi, E.; Kumar, A.; Proença, H. SSS-PR: A short survey of surveys in person re-identification. *Pattern Recognit. Lett.* **2021**, *143*, 50–57. [CrossRef]