City Model Enrichment

Philip D. Smart, Jonathan A. Quinn and Christopher B. Jones

School of Computer Science and Informatics,
Queen's Buildings,
Cardiff University,
5, The Parade,
Cardiff,
United Kingdom
{p.smart, j.a.quinn, c.b.jones}@cs.cf.ac.uk

Abstract

The combination of mobile communications technology with location and orientation aware digital cameras has introduced increasing interest in the exploitation of 3D city models for applications such as augmented reality and automated image captioning. The effectiveness of such applications is, at present, severely limited by the often poor quality of semantic annotation of the 3D models. In this paper, we show how freely available sources of geo-referenced Web 2.0 information can be used for automated enrichment of 3D city models. Point-referenced names of prominent buildings and landmarks mined from Wikipedia articles and from the OpenStreetMaps digital map and Geonames gazetteer have been matched to the 2D ground plan geometry of a 3D city model. In order to address the ambiguities that arise in the associations between these sources and the city model, we present procedures to merge potentially related buildings and implement fuzzy matching between reference points and building polygons. An experimental evaluation demonstrates the effectiveness of the presented methods.

Keywords: Internet/Web, Geometry, Georeferencing, Registration, Modelling, Retrieval

1. Introduction

The availability of 3D models of urban landscape has been improving recently with many models now appearing on open access web sites in addition to the professionally constructed models that might be commissioned for example by city planners. There are several motivations for the production of these models. Historically their main uses have been to assist in visualising landscapes that may be subject to planning and development, in creating virtual tours and games that again are based primarily on visualisation, and for modelling path loss for radio-communications network planning. More recently, with the proliferation of mobile phones and digital cameras, there is increasing interest in the use of 3D models to assist in providing detailed information about an individual's immediate environment through the use of augmented reality (AR) techniques. Using thse techniques, the information and geometry of nearby features can be overlaid on live video or still images of the environment. It has also been shown that enriched models of the landscape can be used to assist in captioning photos, as was demonstrated in the Tripod project (Hall et al., 2009; Smart et al., 2009).

A characteristic of some of the recent applications of 3D models is that their effectiveness depends not so much on visualisation, but upon good quality annotation and attribution of the features of the model so that this information can be associated with images displayed or described to the user. In practice however most of the 3D models currently available are relatively poorly attributed, semantically. It is this paucity of semantic annotation of 3D models that motivates the work presented in this paper.

An obvious way to improve the descriptive data associated with 3D models is to inherit the data associated with 2D city (or other large scale) plans from which they may be been partially derived, or to match them with such plans. Most of the detailed 2D topographic data that includes building outlines come from national mapping agencies and are typically associated with address data. The nature of this address data can of course vary but it is commonly the case that it does not include the identity of the commercial and public agencies that use the buildings, or indeed all of the names of public and cultural buildings, such as museums, castles and churches, that they may represent. There is a need therefore to link to other, additional, sources of semantics. Here we focus upon freely-accessible resources in which descriptors of buildings are associated with geo-references that enable the descriptors to be linked to the corresponding buildings. A prime source of such knowledge is Wikipedia in which there is a large number of geo-referenced articles about buildings, places and other landmarks. Other Web 2.0 resources that we employ are OpenStreetMap and the Geonames gazetteer. It may be noted that information from such sources can itself be ambiguous, as the geo-references can hold various spatial relationships to the features to which they refer. Thus there is a need for methods to resolve these ambiguities.

In this paper we address the target problem of enriching the semantic attribution of 3D models using data mined from Web 2.0 sources. The intention is that the derived attributes will be stored alongside the 3D geometry. It is notable that not all 3D modelling formats support such data. The Keyhole Markup Language (KML) along with most 3D computer graphics formats (VRML, 3D studio Max etc) is only intended to represent 3D objects for visualisation proposes, with no support for detailed semantics of building parts. In contrast, the more recent standard of CityGML (Kolbe et al., 2005) provides an urban landscape ontology that is suitable for representing the geometry of a city model alongside thematic attributes about each building and their parts. Examples of previous work on adding thematic knowledge to 3D models are provided by Kumke (2003) and Hoegner et al. (2007), in which 3D models are matched to facts in underlying official municipal and cadastral datasets. However, to the best of our knowledge, no previous work has attempted automatic enrichment of a 3D city model (in any format) from freely available Web 2.0 information sources.

The paper is organised as follows. In Section 2 we review relevant literature regarding the inheritance of associated semantic data attributes. We also refer to examples of previous work concerned with mining geo-referenced information from the web. In Section 3 we introduce the city model used as an example in this work, and briefly discuss its generation. In Section 4, we describe web-mining methods for the retrieval and attribution of building data, with the aim of producing a highly detailed, accurate, and annotated 3D city model. In Section 5, we evaluate the automated joining algorithm and the accuracy of the semantic enrichment. Finally, in Section 6, we draw conclusions from this work, and discuss future research.

2. Background

There have been considerable efforts to create semantically enriched 3D models. For example Ross et al. (2009) describe the integration of data from multiple sources, including 2D digital map data and terrain models, to create a 3D city model of Potsdam to support processes of urban land management. Similarly Döllner et al. (2006) emphasised the importance of integrating multiple sources of data, in particular semantics in addition to the geometry, to create virtual city models for urban planning. Some aspects of semantic enrichment have focused on the essentially 3D aspects of city models such as the representation of the floors of buildings which will facilitate linking of address data (such as of commercial companies) that may relate to specific parts of the buildings (Döllner et al., 2005) and the representation of facades, in combination with other associated data (Kumke et al., 2007). Directly linked to issues of data integration and semantic enrichment of 3D models has been the exploitation and development of data modelling initiatives from the domain of geographical information systems, resulting in the CityGML (Kolbe et al., 2005) XML vocabulary (itself an application schema of the Geographical Markup Language GML3) and from the domain of building information models (BIM), resulting in the Industry Foundation Classes (IFC) model (Döllner and Hagedorn, 2007). These models provide support for detailed modelling of urban and building structures and provide support for naming and describing building features with information of the sort that may be inherited from 2D digital maps or mined from Web 2.0 sources as described here. The potential of ontologies to support interoperability between 3D urban models has been highlighted by Métral et al. (2009), who show how ontologies can assist in matching concept terms in an application of 3D city models to personal travel planning.

Most work to date on integration of data for 3D city models has employed conventional data sources that are fairly well structured and derive from commercial and public service agencies. The work that we present here differs in its emphasis upon the use of freely available Web 2.0 information sources. A notable recent example of a similar interest in exploiting such data for generation of 3D city models is that of Neubauer et al. (2009), who demonstrate the integration of 2D digital map data including building outlines, roads, land use areas and points of interest from OpenStreetMaps with digital elevation data from the SRTM model. In their work they extruded the OSM building plans with default height values. It should also be pointed out that the focus of the work that we present here is upon the use of Web 2.0 sources that provide essentially 2D data that can be used to enrich the description of 3D models, particularly with familiar or culturally significant names of buildings and landmarks. We do not address here issues of use of such data to provide actual 3D information.

The idea of matching spatial datasets in order to integrate the best aspects of each or to transfer properties of one dataset to another is well established in the field of GIS and is referred to as conflation, most examples of which refer to 2D datasets (e.g. (Samal et al., 1994)). The problem encountered here of matching a 3D model to a 2D map can be reduced to the 2D problem by projecting the building boundaries of the 3D model to the horizontal plane. A variety of techniques have been described to perform the matching of geometry between representations including statistical methods based on mutual information concepts of communication theory (Walter and Fritsch, 1999) and Bayesian maximum likelihood methods (Jones et al., 1999). The use of Web 2.0 resources to discover and link geographically-referenced information has received increasing attention in recent years, though its use to enhance digital map datasets is not well established. An example of linking RDF (semantic web linked data) versions of Wikipedia (DBPedia), Geonames and Flickr is provided in (Schenk et al., 2009), though it was not applied there to enhance other geo-spatial digital models. An example of linking geo-referenced Wikipedia articles to the content of maps on a location-aware mobile application is found in (Baldauf and Simon, 2010).

3. City Model Generation

In this section, we introduce the city model used for the enrichment process. While various georeferenced city models exist, for example in Google Earth, the current accuracy of many of these models, their placement, and level of detail can vary greatly. Manual registration of such models with accurate 2D ground plans can be very time consuming. In addition, some popular modelling languages used to describe city models, notably KML, do not include any detailed semantic attribution about buildings and their parts. Thus, we use the method described in (Quinn et al., 2009), to combine detailed 3D models with accurate 2D city ground plans. In order to make associations between 3D model geometry and the 2D map data and web resources that contain the required semantics, it is necessary to make logical connections between the 2D and 3D geometry and to transfer to the 3D models some of the logical segmentation of the 2D models that corresponds to the presence of individually addressed, or otherwise named, entities - as illustrated in Figure 1. However, while there are many, detailed, hand designed 3D building models available, such models are

Figure 1: 3D scene registered with 2D ground plan data. Also illustrating how names of places from Web 2.0 sources can be integrated into the 3D model once they are associated to an underlaying 2D ground plan. (a) St. Stephan (church), (b) Obere Pfarre (place of worship), (c) Klosterbrau (pub), (d) Battingerhaus (attraction), 3D model is shown positioned above the 2D ground plan for visualisation purposes.

not always robustly structured in that they may suffer from geometric and topological inconsistencies, non-affine transformations, and unknowns due to a lack of design intent knowledge. The city model is generated through the combination of multiple commonly available datasets of the city of Bamberg, Germany. These sources are: A 2D city ground plan, a set of high-quality triangulated 3D models of various cultural or significant parts of a city, a digital elevation model (DEM) of the area, and satellite imagery of the city.

The 2D city ground plan data-set, P, is a set of M buildings in the *DHDN / Gauss Kruger Zone 4* projection, where each building is represented as a single, planar, polygon p_i, and $P : \{p_1, \cdots, p_M\} \subset \mathbb{R}^2$. The dataset P in this work is typical for that available from a city or council for planning applications development, and is assumed to be the most accurate representation of the ground plans of the buildings within the city. The 3D data-set consists of arbitrary groups of buildings from within the city. Each group is assumed to be modelled as a set of polygons in $\mathbb{R}^3$, which may be either connected or disjoint, i.e. no assumption is made regarding which polygons belong to which buildings, or any internal segmentation within a building. The quality of the geometry is not assumed to be good; the topology of the scene is entirely arbitrary, holes may exist, polygons may intersect, be incorrectly aligned, etc. Each scene is therefore treated as the sort of data typically available from user-contributed services such as Google 3D Warehouse, being generally created by hand and not assumed to be created by a professional designer.

The combination of the 2D and 3D models results in a well-segmented set of buildings that conforms to the building segmentation determined by the accurate 2D ground plan dataset P, which is very important for the semantic enrichment discussed in Section 4. The set of buildings is defined as $B : \{b_1, \cdots, b_m\} \subset \mathbb{R}^3$, forming part of the enriched city model C. In order to produce an accurate model of a city, the buildings B_i in the city model C must be projected onto a terrain model. The digital elevation model used in this work consists of a set of height values, registered to the *DHDN / Gauss Kruger Zone 4* projection. Note that no internal building geometry has been incorporated into this model.

4. Semantic Enrichment

In this section we show how to enhance the city model C with thematic information about each building (where available). The techniques developed are general and applicable to any city model which has been registered to some real world coordinate system. To find thematic information for buildings b_i, the polygons $p_i \in P$ (in the 2D ground-plan) are matched to point referenced places (or buildings) in Wikipedia, OpenStreetMaps and the free web gazetteer (a directory of place names with thematic attributes and spatial locations) Geonames. Any found information can then be added to each building b_i and, as the 3D model is registered with the 2D model, enfused into the final enriched 3D model.

Information from geo-referenced Wikipedia articles is extracted as RDF (The Resource Description Framework) triples by the DBpedia project and exposed through a public API. Wikipedia articles are then accessed using the SPARQL RDF query langauge, and stored locally in a Postgres spatial database. Information from Geonames is pre-extracted from their Restful API and again stored in a local Postgres spatial database. OSM locations were taken from Points of Interest OSM extracts in Shapefile format, and again converted to and stored in a local Postgres spatial database.

The complete semantic enrichment process is illustrated is Figure 2. In overview, the 2D ground plan is preprocessed to join component building parts together, and remove non-building parts i.e. walls. This new joined ground plan is then fed into either the baseline fuzzy mapping technique or prominent building fuzzy mapping technique which, in essence, associates locations from OSM, Geonames and Wikipedia to buildings in the 2D ground plan. This mapping is transferred to the 3D model and a semantically enriched 3D model is produced

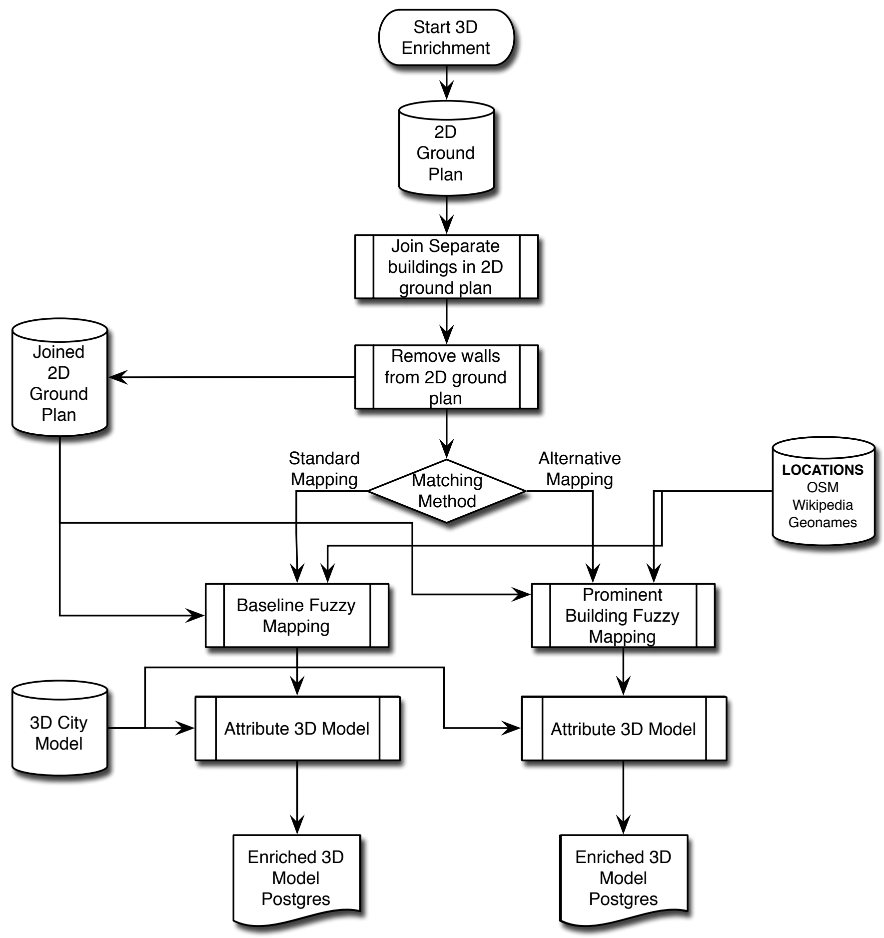

Figure 2: A flow diagram showing the complete semantic enrichment process. From input of the 3D model, 2D ground plan and store of locations, to output of a semantically enriched 3D model in a Postgres spatial database

(Stored in a Postgres spatial database),where of interest this information could then later be used to produce a semantically enriched model in CityGML format - although such work is outside the scope of this paper. Each of these processes is described in more detail in the sections to follow.

4.1. Preprocessing the Ground Plan

The Bamberg 2D ground plan P has 3665 unique building objects. However, as indicated earlier, often buildings that represent a semantically distinct entity, are segmented into a number of different building polygons. For exam-

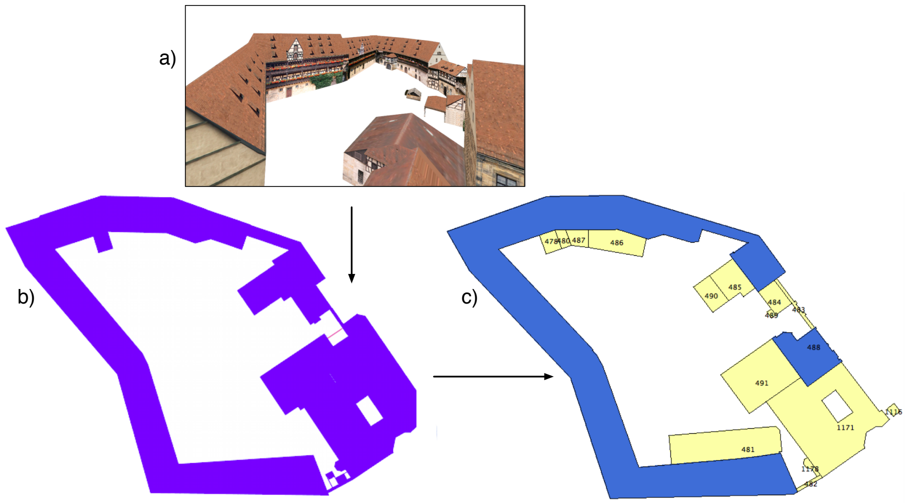

Figure 3: Various models of the Alte Hofhaltung building: a) Detailed 3D model, b) Top down orthographic projection, c) A set of buildings in the 2D ground plan P, all of which represent a part of the building

ple, Figure 3 shows both the orthographic top down projection of the 3D model that represents the Alte Hofhaltung (illustrating its 2D footprint), and the set of buildings polygons from the ground plan P that represent the same building entity. Each of these building polygons should be linked to the same information resource. This situation is characteristic of the ground plan. Consequently, a preprocessing method that groups together segmented buildings is required, such that during linking from point referenced information sources to buildings in P, the correct extents of these semantically distinct buildings are mapped onto.

In this section we develop a general preprocessing method for building grouping based on building connectivity and distance. The method groups buildings without addresses (shown in yellow in the 2D ground plan in Figure 3) to those buildings with addresses (shown in blue in Figure 3) under the assumption that buildings with an unique address should be preserved, and only joined to those closely connected buildings without addresses. The grouping forms a new ground plan P_{join} based on connectivity (preserving address boundaries) and address similarity.

4.1.1. Preprocessing Implementation

Building parts without addresses denoted B_{addr-} (where $B_{addr-} \subset P$) are joined to buildings with addresses denoted B_{addr} (where $B_{addr} \subset \{P \backslash B_{addr-}\}$) by considering building connection, creating the joined set P_{join}. The result of preprocessing is to merge or union the geometry of individual building polygons without addresses, with those building polygons with addresses, ideally in a way such that the resulting 2D building outline is a better representation of the corresponding building outline of the registered 3D buildings (generated as described in section 3).

Joining Buildings by Connection. Distance (building proximity) and connectivity are the principle heuristics used to join buildings. That is, buildings are joined based on which are connected, how close they are to each other and the degree of connection. The first step involves finding all paths between each building $b_i \in B_{addr-}$ (without address) to each building $g_i \in B_{addr}$ (with address) using a function denoted `CountHops`.

`CountHops` counts the number of *hops* from the input building b_i to each connected building g_i. A *hop* is a traversal from one building to an adjacent building with which it shares the whole or part of a boundary edge. Figure 4 illustrates the number of hops between the building with ID 3136 and the nearest connected buildings from B_{addr}. Each complete set of hops (a path) identifies a connected addressed building.

At this stage the path with least *hops* is not chosen for grouping. To illustrate why, note that in the example in Figure 4, building 3136 is only 2 hops from building 1379, whereas it is 3 hops from its more obvious grouping to building 1378. Hence, once all possible paths (a set denote Ps) have been generated for a building b_i, two measures for each path are evaluated and optimised. The first is a measure of connectedness between each of the buildings in a path. That is the mean length (m) of the intersecting (shared) edges (a function denoted El). The second measure is the total of the Euclidean distances between centroids of each adjacent unaddressed building in the path, plus the distances from centroids to neighbouring edges of the addressed building g_i (termed path length, a function denoted Pl). Examples of both measures are shown in Figure 5.

For each path $Pc_i \in Ps$, the total intersection length and centroid distance is computed. We then seek to find solutions that minimise the total centroid distance and maximise the total intersection length. The optimum solution is chosen by maximising the average mean function Best(Pc_i) as applied to all candidates Pc_i in the set of all possible paths Ps.

$$Best(Pc_i) = \frac{2(1 - Pl(Pc_i)) + El(Pc_i)}{3} \tag{1}$$

Pl is maximised in function Best, and Pl has twice the weighting of El as, during empirical evaluation, it was discovered best to favour smaller path lengths over larger edge intersection lengths. As an example, in Figure 4 building 1378 was joined to building 3136.

Once a best solution is obtained using the function Best, the start building b_i and end building g_i are stored as a tuple $\{b_i, g_i\}$ and added to the *join* stack JS. This process is repeated for all starting buildings b_i, and once all buildings b_i have been joined to a building g_i, the join stack JS is traversed and the polygonal geometries of each building tuple $\{b_i, g_i\}$ are merged. This operation

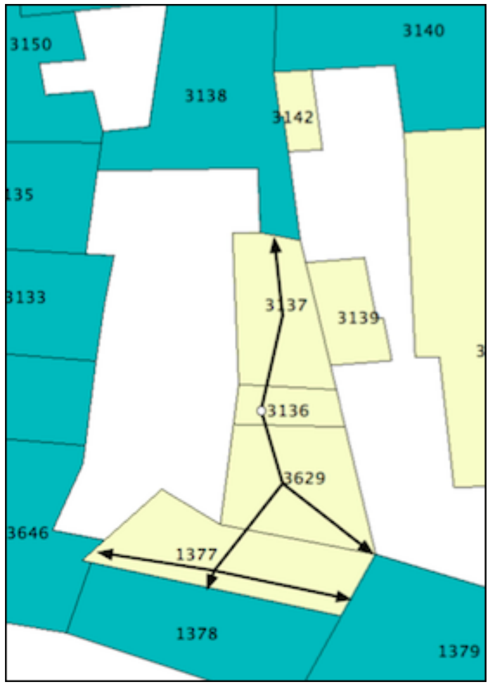

Figure 4: All possible paths (including each hop) between the building 3136 and buildings with address information (in blue)

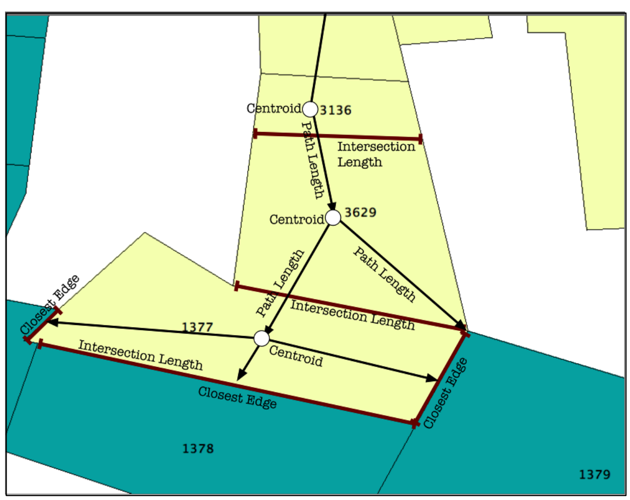

Figure 5: Example intersection length and centroid distance

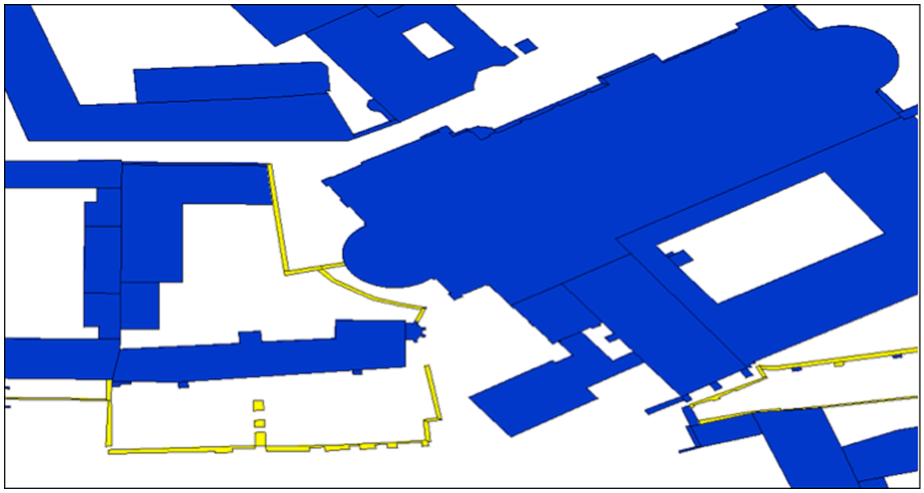

Figure 6: Polygons representing buildings connected through polygons representing city walls (in yellow)

Figure 7: Example city walls from the 3D model C

creates the set P_{join} of joined buildings. Thus P_{join} then contains the identities of addressed buildings, the corresponding geometry of which has been updated to include the associated previously unaddressed building objects with which they have been merged.

4.1.2. Improving Building Joining

As shown in Figure 6, some buildings are joined through city walls - examples of city walls from the 3D model are shown in Figure 7. One option for improvement when joining buildings is to omit hops across city walls, which can have the effect of joining buildings that should not be directly connected. However, in order to prevent this the system must be capable of first identifying polygons that represent city walls.

As wall objects are non-compact elongated shapes, shape description vectors

($\mathbf{S}(b_i)$) for each building shape b_i in P_{join} are constructed based on the notions of elongation and compactness. However these two descriptors are scale invariant and, on their own, are not enough to distinguish between elongated wall shapes and elongated building shapes. Hence a third descriptor is introduced into $\mathbf{S}(b_i)$ based on object area which helps to distinguish elongated non-compact buildings, as they have much larger area than walls. Each shape descriptor is now described, followed by a definition of the final shape description vector.

The elongation measure of a shape b_i with polygonal boundary is defined in (Stojmenović and Žunić, 2008) as:

$$\epsilon(b_i) = \frac{\sum_{1\leq i\leq n}|e_i| + \sqrt{\left(\sum_{1\leq i\leq n}|e_i|cos(2\alpha_i)\right)^2 + \left(\sum_{1\leq i\leq n}|e_i|sin(2\alpha_i)\right)^2}}{\sum_{1\leq i\leq n}|e_i| - \sqrt{\left(\sum_{1\leq i\leq n}|e_i|cos(2\alpha_i)\right)^2 + \left(\sum_{1\leq i\leq n}|e_i|sin(2\alpha_i)\right)^2}}$$

where e_i $(1 \leq i \leq n)$ are edges of the boundary of buildings b_i, and α_i $(1 \leq i \leq n)$ are angles between the edges e_i and the x-axis. Given a shape b_i of perimeter P and area A, the compactness measure for shape b_i is defined as (see for example (Lee et al., 2004)):

$$\mathbf{C}(b_i) = \frac{P^2}{A}$$

A shape description vector $\mathbf{S}(b_i)$ of a building b_i is then defined by its elongation measure, compactness measure and area:

$$\mathbf{S}(b_i) = \begin{pmatrix} \epsilon \\ C \\ A \end{pmatrix}$$

In order to create a classifier for walls, a training set of 20 walls, 20 terraced buildings, 20 small building parts and 20 large building shapes was taken from P (0.016% of all building shapes), and their shape vectors stored as the set Ls. Example training polygons representing walls, and those representing large prominent buildings e.g. places of worship, museums etc. and terraced buildings are shown in Figure 8

During building joining, a shape description vector is generated for each start building b_i and connected building g_i, and matched against each of the shapes l_i in the set of learned shape descriptors Ls using the cosine similarity measure, which produces a value in [-1,1], where the highest value is taken as the closest match:

$$similarity(b_i/g_i, l_i) = \frac{\mathbf{S}(b_i)\cdot\mathbf{S}(l_i)}{||\mathbf{S}(\mathbf{b_i})||\ ||\mathbf{S}(l_i)||}$$

If a shape b_i is classified as a wall, the algorithm moves on to the next start building $b_i \in B_{addr-}$. If the shape g_i is classified as a wall, the algorithm will not consider it a possible *hop* and hence will ignore it as a possible join path.

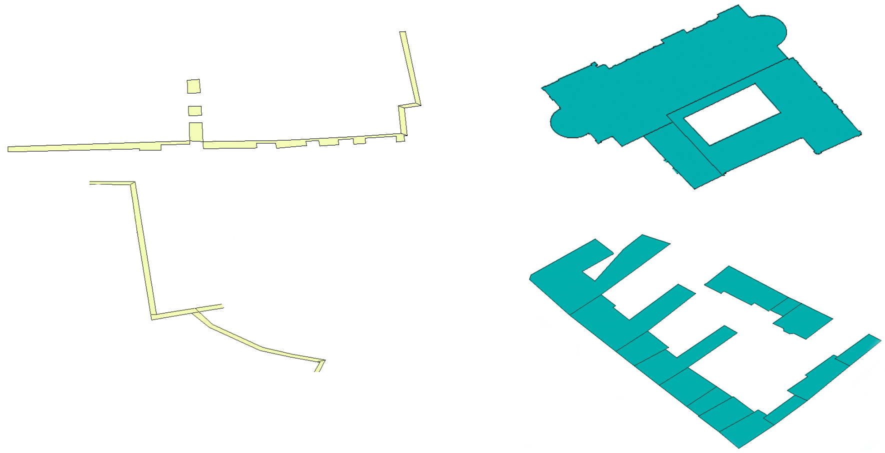

Figure 8: Example wall (left) and building (right) shapes

4.1.3. Preprocessing output

After running the complete preprocessing steps over the input ground plan P, a new ground plan P_{join} is created as shown in Figure 9. One immediate observation is that most 'wall' shapes have been removed, as they were not considered start buildings b_i. In addition any building without address that was not physically connected to buildings with addresses is also not contained in P_{join} . Removing walls and non-connected buildings helps to remove noise from the ground plan, making matching points in v more robust as shown in the evaluation section. Algorithm 1 describes the complete pre-processing algorithm.

4.2. Matching Issues

Matching building polygons $b_i \in P_{join}$ to point referenced locations from each of the three sources ranges in complexity from simple containment queries (point in polygon checking), to non-trivial cases that involve mapping a single point referenced location to a number of spatially disjoint buildings. These cases, in order of increasing complexity, are now described. From this point onward, the complete set of point referenced building locations ν from Wikipedia / DBpedia (W), Geonames (G) and Open Street Maps (O) is defined as:

$$\nu = \{W, G, O\}$$

Case 1 - Direct Containment. In the simplest case, point referenced locations of buildings in ν lie directly inside a polygon b_i. Hence, a simple point in polygon match is applicable, see for example the Cathedral de Bamberg example in Figure 10(a).

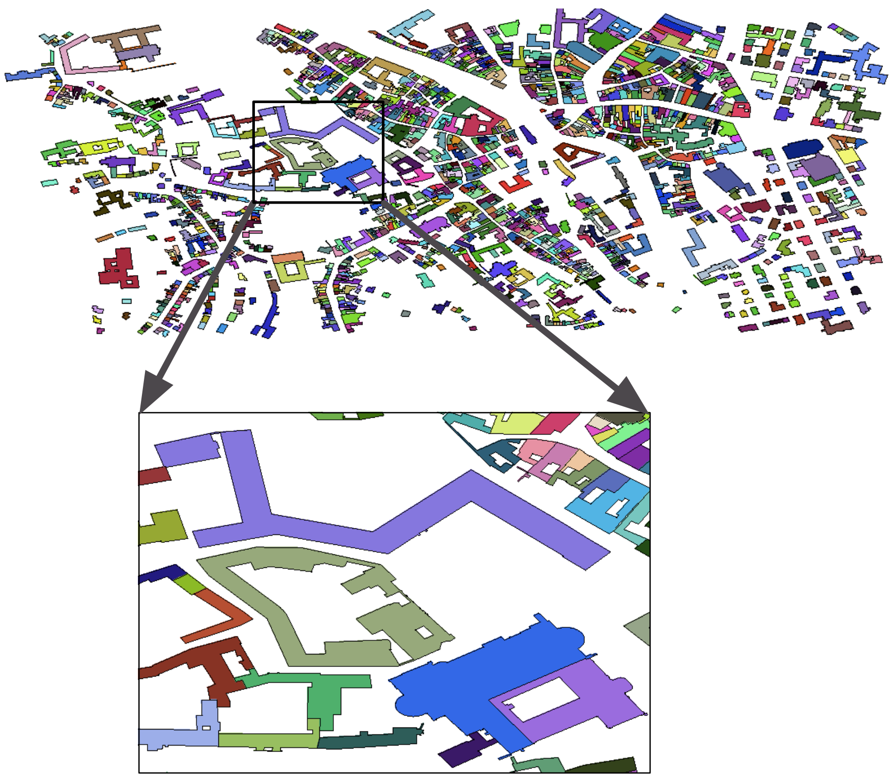

Figure 9: The final joined ground plan P_{join}

Case 2 - One to One Matching. Point referenced locations of buildings in ν do not always lie inside a polygon b_i in P as in case 1. Figure 10(b) shows how the Alte Hofhaltung Wikipedia article has been geo-referenced outside of its actual building polygon.

Case 3 - One to Many Matching. In a slight alteration to case 2, certain locations in ν can refer to a number of separate buildings. Figure 10(c) shows a Wikipedia reference that refers to a block of 17 separate buildings referred to as part of *Small Venice, a former Fischer settlement on the eastern bank of the river Regnitz in Bamberg.*

Case 4 - One to One With Many Disjoint Buildings. A single building location in ν may map to more than one disjoint building b_i in P that belongs to the same set of buildings. For example, University of Bamberg is comprised of 4 spatially disjoint buildings in this area of the city, see Figure 10(d).

Outcomes:. Case 1 is easy to solve with a point in polygon check. Clearly it is possible that, even though the point location is contained inside a single

Algorithm 1 Preprocess()

```
Let JS be the join stack, which is a set of building tuples e.g. the building
without address, and the best connected building with address
Let shapeVectors[] be the training set of shape vectors representing walls,
terraced buildings, small buildings and large buildings
Let P be the set of polygons in the 2D ground plan
for Polygon b ∈ P do
  if b does not have an address then
    match b to set of shapeVectors using cosine similarity measure
    if b is not classified as a wall shape then
      Let allHops[] = countHops(b, P, shapeVectors)
      if sizeof(allHops) == 1 then
        add b and last connected polygon entry in allHops[1] to JS
      else
        Let bestHop = compute best chain by choosing the best chain from
        allHops[] using the function Best (function 1)
        add b and last connected polygon entry in bestHop to JS
      end if
    end if
  end if
end for
Join building geometries in JS
```

building, it actually maps to more than one building. However, our approach aims to be conservative, where the precision of mapping is more important than recall. In other words, we would rather map to one correct building, than map to many buildings where only a subset are correct. Case 2 could be solved by finding the nearest (in terms of Euclidean distance) building, however in some cases, the nearest building may not always be the correct building - see Figure 11, where building 126 is closer than its correct mapping, building 151. Cases 3 and 4 are, with the information we have, non-trivial. Case 3 would require associating 17 buildings to the same Wikipedia article, while not associating any of the other, often still connected, buildings. An obvious approach here would be to use the registered 3D model to consider occlusion. However, from manual investigation it appears users do not themselves consider occlusion when tagging articles in Wikipedia, or when adding locations in Open Street Maps or Geonames i.e. some references are to occluded buildings, hence this approach would not apply to all cases. Case 4 is a specialisation of case 3, which also requires associating a number of nearby buildings to a single Wikipedia article, but here each member of the spatially disjoint set of buildings is semantically part of the same building entity (i.e. the set of buildings is referred to by a single name).

In this paper we develop two fuzzy mapping functions that associate, to a certain degree, each georeferenced information source (the points in ν) with

Algorithm 2 countHops(Polygon buildingPolygon, GroundPlan P, ShapeVectors shapeVectors[])

```
Let allHops[] be the set of all possible join paths
METHOD count(Polygon buildingPolygon, currentHops[], VisitedPolygons[])
for Polygon b ∈ P do
  if b is not in visitedPolygons then
    if b intersects with buildingPolygon then
      match b to set of shapeVectors using cosine similarity measure
      if b is not classified as a wall shape then
        if p has an address then
          add b to currentHops
          add currentHops to allHops[]
        else
          add b to currentHops[]
          add b to visitedPolygons[]
          count(p, currentHops, visitedPolygons)
        end if
      end if
    end if
  end if
end for
END METHOD
return allHops[]
```

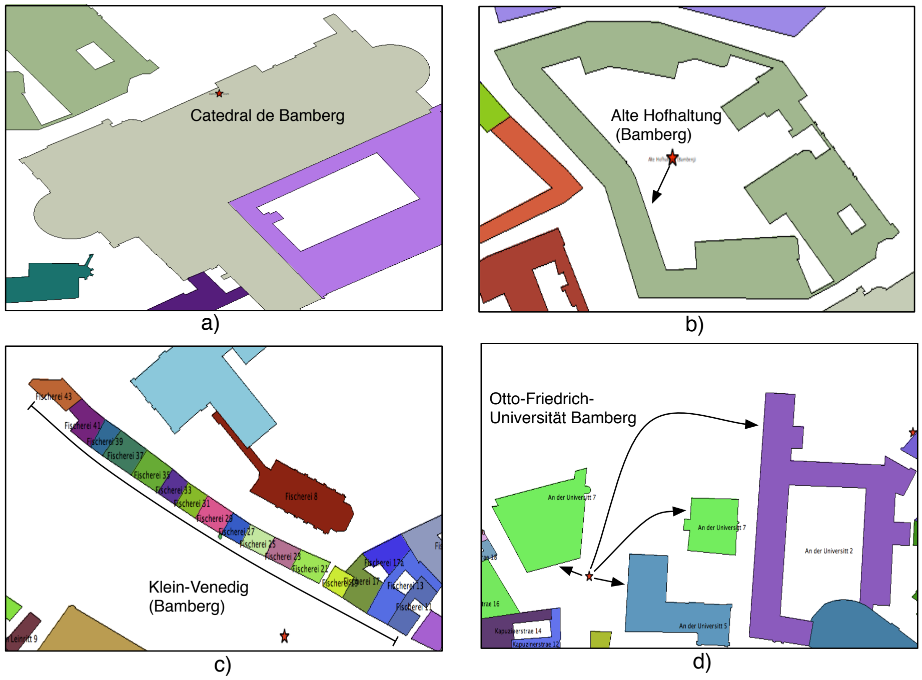

Figure 10: Example mappings between Wikipedia georeferenced articles and buildings in P

buildings in the 2D ground plan P_{join}. By using a fuzzy mapping we hope to overcome some of the issues previously described. The fuzzy mapping functions are described in the sections to follow.

4.3. Fuzzy Mapping Function

In this section we describe the process of linking points in ν to building polygons b_i in P_{join} using one of two different fuzzy mapping functions. Once linked to the joined 2D ground plan, this information can be added to the registered 3D city model C.

Here we use the notion of fuzzy relations (Zadeh, 1965) to map locations to buildings. More specifically, we use a fuzzy relation $\mathbf{R} : \nu \times P \mapsto [0,1]$ to map points from ν to buildings in P, where the degree of truth in [0,1] to which the mapping holds is determined using two fuzzy mapping functions $\mathbf{R}$ as described in sections 4.3.1 and 4.3.2. The fuzzy relation $\mathbf{R}$ forms a new fuzzy set Ω, which is a list containing element and membership degree pairs; $\Omega = \{\{x, y\}, \mathbf{R}_1\}, \{\{x, y\}, \mathbf{R}_2\}, \ldots, \{\{x, y\}, \mathbf{R}_{nm}\}$, where x is a point from the set ν, y is a building from the set P, n is the size of the ν and m the size of P, and $\mathbf{R}$ is their membership degree in [0,1], e.g {{Cathedral de Bamberg, 1191},1}, {{Alte Hofhaltung, 479},0.9}, {{Alte Hofhaltung, 495},0.87}}.

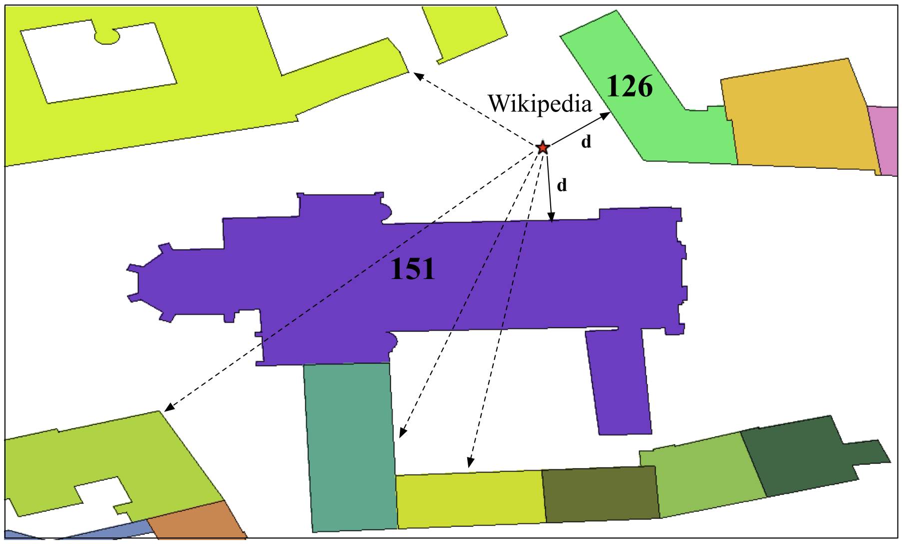

Figure 11: Possible inaccuracies of nearest building match (St. Jakob). The dotted lines represent example distances between the Wikipedia location for St. Jakob and some of the buildings in the ground plan P

4.3.1. Baseline Fuzzy Mapping

The Euclidean distance, d, from point locations in ν to building b_i polygons in P_{join} is based on the distance from the point to either the nearest edge of the building or to the nearest vertex of the building, depending on which is closer. The baseline fuzzy relationship $\mathcal{R}$ between a point p_i from the set ν, and building b_i from the set P_{join} is then computed using a normalised distance measure in [0,1]. That is, by normalising the computed distance d against the maximal distance between the point p and all buildings b_i in P_{join} :

$$\mathbf{R}(p,b) = 1 - \left(\frac{d(p,b)}{d_{max}(p)}\right)$$
$$d_{max}(p) = max_{b_i \in P_{join}}(d(p,b_i))$$

Relations closer to 1 represent better mappings. All directly contained points (points that lie inside building polygons) have a distance of 0 and hence a degree of membership $\mathbf{R}$ of 1.

Considering multiple evidence across sources: . Many points in ν may link to the same building in P_{join}. A many-to-one linking can be added as extra evidence for the fuzzy relation $\mathbf{R}$. For example, Figure 12 shows both the Wikipedia point reference and Open Street Maps point reference to the same building (building 151 or St. Jakob's church).

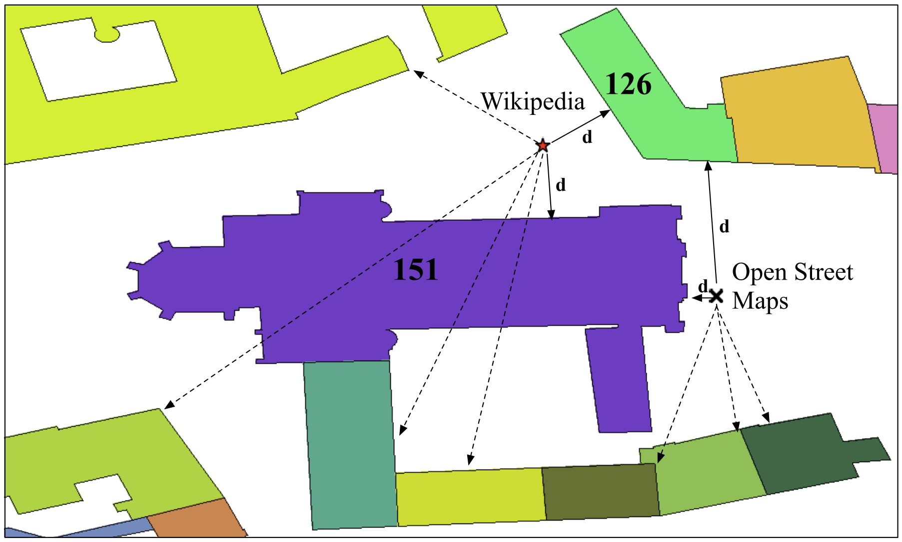

Figure 12: Improving matching by considering multiple evidence (St. Jakob). The dotted lines represent example distances between each location and some of the buildings in the ground plan P_{join}

Consequently, the normalised distance fuzzy relationship function is extended to include mappings that consider more than one identical point reference. To identify identical point references in ν, standard and alternative names of each article or POI are matched, using a combined soundex and edit distance fuzzy string similarity measure. Sets of identical references $\nu_1 = \cdots = \nu_n$ are then removed from ν and added to a new set $\nu^=$ as tuples $t = \{\nu_1, \cdots, \nu_n\}$ where, for the set $\nu^=$, $n \geq 2$. For simplicity, we also assume from this point onward that remaining elements of ν are actually tuples t but with only one element i.e. $n = 1$. Hence the total combined set of point references V is then formed from entries in ν and entries in $\nu^=$ e.g. $V = \nu^= \cup \nu$.

Once V has been established, membership degrees $\mathbf{R}$ in Ω relating point references in V to buildings in P_{join} are computed using the normalised mean distance between each evidence point ν_i in a tuple t_i in V (where $0 < i < |t_i|$), and each building. More formally, the relation $\mathbf{R}$ is computed for a tuple t in V and a building b in P_{join} as:

$$\mathbf{R}(t, b) = 1 - \left(\frac{d(v_i, b) + \cdots + d(v_n, b)}{d_{max}(t) \times n} \right) \quad , \text{where } n \geq 1$$

Where in this case d_{max} is used to find the maximum distance between the mean distance of points in a tuple t and all buildings b_i in P_{join} e.g.

$$d_{max}(t) = max_{b_i \in P_{join}} \left(\frac{d(v_1, b_i) + \cdots + d(v_n, b_i)}{n} \right)$$

An element membership degree pair $\{\{x, y\}, \mathbf{R}\}$ is then added to the set Ω for each tuple t_i and building b_i by taking the computed degree as the element $\mathbf{R}$, the point v_i with minimum distance as element x, and the building b_i as the element y.

Finally, the fuzzy relation function $\mathbf{R}(t, b)$ can be further improved by introducing a common sense heuristic that point references in V will not be over 100m from the building(s) they refer to. Hence the final baseline function $\mathbf{R}_b$ returns a membership degree of 0 for all those buildings outside a 100 metre radius from the minimum distance point v_n in a tuple t_j:

$$d_t(t, b) = \frac{d(v_1, b) + \cdots + d(v_n, b)}{n}$$

$$\mathbf{R}_b(t, b) = \begin{cases} 1 - \left(\frac{d_t(t,b)}{d_{max}(t)}\right) & \text{if } d_t(t, b) < 100 \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

where $n \geq 1$.

4.3.2. Prominent Building Fuzzy Mapping

We have also developed an alternative fuzzy membership function which boosts degrees of membership for prominent building shapes. A building classifier was built to detect prominent buildings. Again we build a shape description vector $\mathbf{S}$ for a sample set of buildings from the ground plan P_{join}, based on the building's elongation ϵ (see (Stojmenović and Žunić, 2008)), compactness C (see, for example (Lee et al., 2004)) and area A (scale). More formally, a shape description for a building b_i is the vector:

$$\mathbf{S} = \begin{pmatrix} \epsilon \\ C \\ A \end{pmatrix}$$

Shape vectors were learnt for 20 buildings in the ground plan that represented walls, 20 that represented terraced or small buildings and 20 that represented large prominent buildings (0.016% of all buildings in P_{join}). The set of learned shapes were added to the training set Lv. The cosine similarity measure was used to match and classify shape vectors for new building shapes against those in Lv.

The new fuzzy membership function denoted $\mathbf{R}_{alt}(t, b)$ is then a function of both distance and building shape. More formally, for a tuple t in V and building b in P:

$$\mathbf{R}_{alt}(t, b) = \begin{cases} \dfrac{W\left(\frac{d_t(t,b)}{d_{max}(t)}\right)^2}{W_{max}(t)} & \text{if } Pb(b) = \text{true} \\ \dfrac{\left(\frac{d_t(t,b)}{d_{max}(t)}\right)^2}{W_{max}(t)} & \text{if } Pb(b) = \text{false} \end{cases} \tag{3}$$

where $d_t(t, b) < 100$, and $n \geq 1$. Note that if $d_t(t, b) \geq 100$, $\mathbf{R}_{alt}(t, b) = 0$.

$Pb(b)$ is a function that takes a building b_i and determines if it is a prominent building by matching its shape description vector against the set of learned shape vectors Lv (as discussed previously). W_{max} is the maximum value (for normalisation) taken from the mean distance of a tuple of points t from V and all buildings b_i in P_{join} squared and multiplied with a weighting W if the building b_i is a prominent building. Algorithm 3 describes the matching algorithm with reference to Equations 2 and 3.

Algorithm 3 ProduceMapping()

Let P_{join} be the set of polygons in the 2D ground plan
Let ν be the set of point locations from OSM, Geonames and Wikipedia
construct the set $v^{=}$ of tuples of identical point locations
Let Ω be the set of fuzzy mappings
for location tuples $t \in \nu^{=}$ **do**
  for Polygons $b \in P_{join}$ **do**
    Let D be a set of {polygon,location,distance} tuples
    for location v in t **do**
      if v lies inside b **then**
        add the tuple $\{v,b,0\}$ to the set of tuples D
      else
        Let distance = the minimum distance between b and v
        add the tuple $\{v,b,\text{distance}\}$ to the set of tuples D
      end if
    end for
    Let D_{avg} = compute average distance for the set D and add new tuple $\{v,b,\text{avgDistance}\}$
    Let maxD = the maximum distance between location v and all Polygons $b \in P_{join}$
    if the desired output is the baseline mapping **then**
      compute the normalised baseline fuzzy relations for D_{all} using Equation 2 and add them to the final set of mappings Ω
    end if
    if the desired output is the alternative mapping **then**
      compute the normalised baseline fuzzy relations for D using Equation 3 and add them to the final set of mappings Ω
    end if
  end for
end for
return Ω

5. Evaluation

In this section, we give an experimental evaluation of the semantic enrichment technique by firstly analysing the accuracy of the building joining preprocessing technique, and then by comparing the accuracy of location-building mappings produced by the automated semantic enrichment process with a set of assumed accurate location-building mappings produced by a human expert (see Sections 5.1 and 5.2).

5.1. Evaluation of Automated Building Joining

As a first step in evaluating the semantic enrichment process, we analyse the accuracy of the automated building joining procedure by comparing it against groupings defined by a human expert. The human expert was given the original ground plan P and asked to join buildings that they thought part of the same building entity. In this way we can compare the expert grouped buildings against the machine grouped buildings in P_{join}.

For each expert building group, the number of building groups identified by the machine to cover the same set of buildings was determined. A building agreement ratio BA is then calculated for each individual expert building grouping as ratio of the number of machine groups per human group e.g. $BA = \frac{1}{NM}$, where NM is the number of machine groups. If the number of machine groups used to group a certain set of buildings is the same as the number of human groups then $BA = 1$. If for example there were two machine groups used to represent one expert grouping, then $BA = 0.5$. The graph in Figure 13 shows BA ratio ranges on the x-axis against matching percentages on the y-axis. Each ratio is bucketed into intervals of 0.1 from [0,1), with a final bucket representing the range [1,1] - to show exact matches.

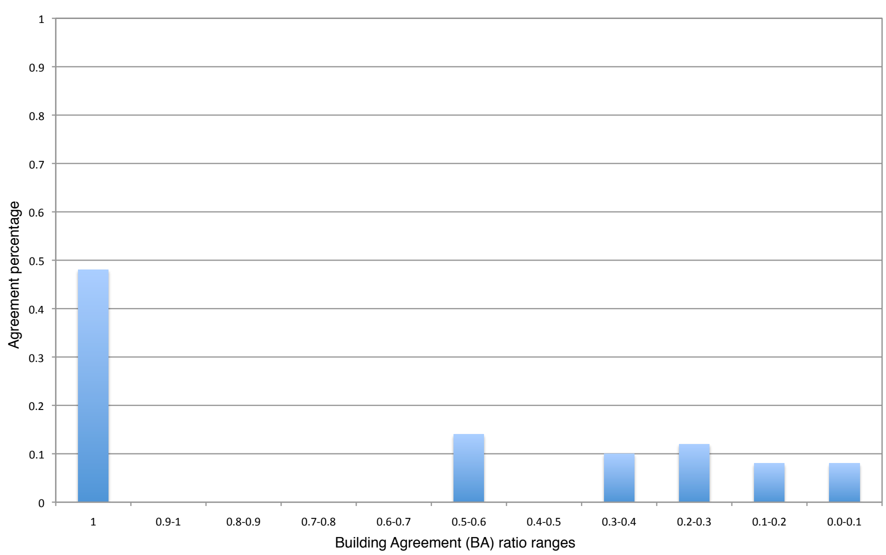

Figure 13: Evaluation of the accuracy of automated building grouping

As shown the machine and expert groupings have a high exact agreement ($BA = 1$) of 48%, thus 24 out of the total 50 expert groups were exact matches.

However, the percentage drops from ratios of 0.6 to 0.0 for the remaining 52% of building groupings. That is, the remaining 26 expert groups were represented on average by 1.6 up to 10 machine groups. Despite this relatively high number of exact ratio agreements, the number of low ratio agreements is a concern. In understanding this, the human expert did not try to group buildings to addressed buildings, instead using only local knowledge to group buildings. Thus the human expert has often grouped buildings with separate addresses - buildings that would not be grouped by the automatic, address preserving, machine grouping method.

5.2. Semantic Enrichment Mapping Accuracy

We evaluate both standard and alternative fuzzy relations applied to the grouped ground plan P_{join} along with the original ground plan P. The original ground plan P is used to see whether building grouping removed noise and increased the accuracy of the linking process. The four different outputs of the system are then:

1. Matching buildings in P_{join} to point referenced locations using the alternative (Prominent) fuzzy membership relation
2. Standard matching of buildings in P_{join} to point referenced locations using the standard fuzzy membership relation
3. Matching of buildings in P to point referenced locations using the alternative (Prominent) membership fuzzy relation
4. Standard matching of buildings in P to point referenced locations using the standard fuzzy membership relation

For the area of Bamberg, the set V has 53 tuples with only one evidence location v, and 10 tuples with evidence from multiple sources i.e. v_i where $i > 1$. After applying both standard and alternative fuzzy mappings over the set P_{join}, the fuzzy set Ω holds mappings between places referenced by articles in Wikipedia, or entries in Geonames and Open Street Maps, and buildings in P_{join}. For evaluation, we compare the results of the mapping after applying different thresholds on the fuzzy relationship $\mathbf{R}$, with manual mappings held in a set ψ as defined by a local expert.

For comparison, we first partition the sets Ω and ψ such that each partition ω_i of Ω represents information about a single unique reference v in V, and similarly for each partition ϕ_i of ψ. For each identical partition ω_i and ψ_i (identical in that they are about the same reference v in V) we then compute the following measures for both baseline (standard) and alternative fuzzy relation functions at different threshold levels of $\mathbf{R}$. The first (represented by the columns in Figure 14) is a measure of the number of exact matches between the machine and expert output. That is, how many of the locations have been mapped exactly onto the same number of buildings as the human expert, without mapping onto other incorrect buildings. The second (represented by the line graph in Figure 14) shows the average (for all v) per threshold level of a combined measure C of

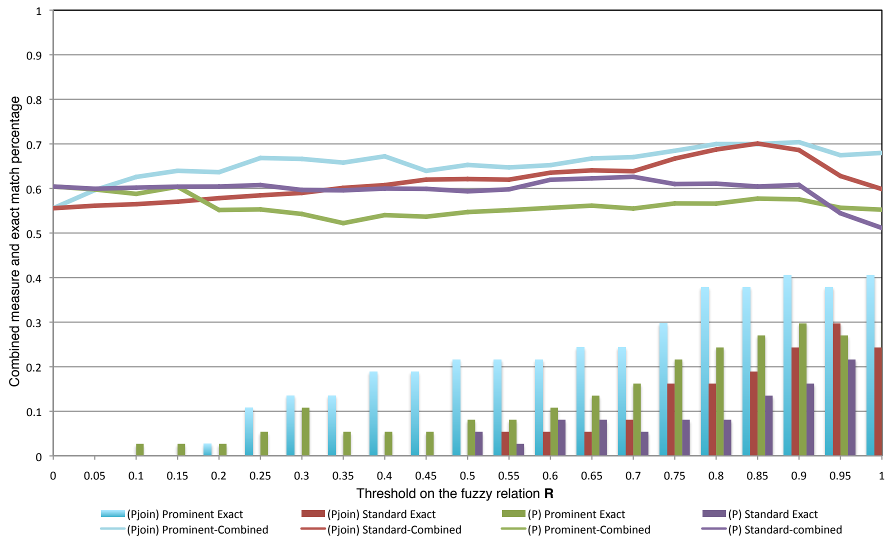

Figure 14: Comparison of the baseline (standard) and prominent fuzzy membership functions

mapping accuracy for each unique point location v. More formally, C is defined as:

$$C(v) = \frac{3(1 - NFP(v)) + (1 - NOP(v)) + 2(NA(v))}{6}$$

Agreement A is a count of the number of buildings correctly matched in the machine output with those from the expert output. Agreement is then normalised (NA) by dividing A by the total number of buildings in the expert output for that partition ϕ. Normalised False Positives (NFP) is the count of the number of buildings linked to point references in the machine output that are not contained in the human output, divided by the total number of machine buildings in ω_i. Normalised Omitted Positives (NOP) is the count of the number of buildings linked in the expert output that are not contained in the machine output, divided by the number of buildings in ϕ. Weightings are introduced such that priority is placed on maximising agreement and minimising machine false positives NFP. This is because, it is assumed better to not match all buildings in the machine output to the expert output and have a low number of machine false positives, than match to all buildings in the expert output but also many others not in the expert output - giving erroneous linkage from buildings to point reference information sources

The maximum score for the prominent (0.6043) fuzzy mappings for the original ground plan P occurs at a low threshold of 0.0. At this threshold the average percentage of FP is 0.93, which equates to 118 buildings (for each location on average) being mapped to a location that is not in the expert output. The maximum score of 0.6261 for the standard fuzzy mapping for the original ground plan P occurs at a higher threshold of 0.7. This is an improvement over the prominent mapping, however this still equates to 8 buildings (for each location on average) being mapped to a location that is not in the expert output. The prominent fuzzy mapping shows a far higher accuracy when considering only exact matches. An exact match percentage of 0.02 starts at a relatively low threshold of 0.1, and continues to improve until hitting a peak of 0.3 at a threshold of 0.9.

The accuracy trend of both standard and prominent fuzzy mappings to buildings in the original ground plan P are below those for mapping to the grouped ground plan P_{join}. Furthermore, the maximum combined score for each is roughly 0.1 (10%) lower than the maximum combined score for each fuzzy mapping method over the grouped ground plan. This lower level of accuracy helps justify the need to group buildings together, in addition to the initial requirement of better matching 3D building footprints. Grouping sets of buildings together helps to remove noise created by large groups of small buildings that have an adverse effect on the normalised distance measure.

Both the prominent and standard fuzzy mappings over the grouped ground plan P_{join} follow a general trend where increasing the threshold increases the combined measure. This is largely attributed to the sizeable decrease in average FP (from approximately 46 to 0.2) for these thresholds compared to comparatively smaller increase of average OP (from 1.8 to 3.4), and small decrease of average A. However, all fuzzy mappings see decreases in combined measure beyond a 0.85-0.9 threshold. At these thresholds the average FP does not drop much, but the average OP is still increasing and the average A is still decreasing. The prominent fuzzy mapping has a marginally better maximum combined measure of 0.704 at a threshold of 0.9, compared to the maximum combined measure of 0.701 at a threshold of 0.85 for the standard mapping. Furthermore, at this threshold the prominent mapping has a 0.4 (40%) exact match success rate, compared to a maximum of 0.3 (30%) for the standard mapping. Indeed the prominent mapping over both P and P_{join} provides a far better exact match rate than the standard mapping. At the best threshold for the prominent mapping, the average number of NFP is 0.18, which equates to a relatively low 0.29 extra buildings being mapped in the machine output. The average NOP is 0.37 or an average 3.29 buildings in the expert output that were not in the machine output, this is an increase from the best value of 0.14 (1.8 buildings) at a threshold of 0.0. However, as previously stated, we prioritise minimising the average NFP over the average NOP.

Consequently, to achieve the best mappings between location and buildings the alternative mapping $\mathbf{R}_{alt}$ at a threshold of 0.9 over the buildings in P_{join} should be used. As a result, buildings in P_{join} are then linked to information sources and output as an enriched ground plan P^E . Buildings in the 3D city model C that are registered to the ground plan P^E inherit the corresponding linkage to information sources.

6. Conclusions and Future Work

In this paper we have presented methods to assist in improvement of the semantic annotation of city models. The main contribution of this paper is the semantic enrichment from geo-referenced Web 2.0 sources. The presented semantic enrichment procedures exploit georeferenced information from three Web 2.0 sources: Wikipedia, Geonames and OpenStreetMap. Matching the georefenced point locations from these sources to the geometry of the buildings in the registered 3D model has entailed several challenges. These relate to the facts that: some individual buildings in the 2D plan may not have address data; the source point references may be either inside or simply near to the buildings to which they refer; the references may refer to a single building object or to several; and the different sources may have different perhaps conflicting locations for the point references. These problems have been addressed through a mix of methods that include merging unaddressed buildings with adjacent addressed buildings, using various heuristics to make this a sensible process; and the development of a fuzzy matching technique that takes account of the proximity of multiple point references from the different sources and the identification of potentially important building geometry that can be expected to have been referred to by name in the Web 2.0 sources. The evaluation of the procedures, using expert knowledge, has demonstrated that they provide what should be a useful contribution to automation of the semantic annotation problem.

Further work regarding semantic enrichment will focus on improving the reliability of the methods, for example through the use of machine learning approaches to the linking of georeferences to buildings, and the extraction of finer level semantic annotations relating to the structure of individual buildings, such as towers and wings of buildings as well as notable features such as historic doors, windows and clocks. Other work will focus on development and generalisation of the model registration process.

ACKNOWLEDGEMENTS

The research reported in this paper is part of the project TRIPOD supported by the European commission under contract No. 045335. We are grateful to Christian Matyas for assisting in the generation of training data.

References

Baldauf, M., Simon, R., 2010. Getting context on the go: Mobile urban exploration with ambient tag clouds. In: Purves, R., Jones, C. B. (Eds.), GIR'10: Proceedings of the 6th Workshop on Geographic Information Retrieval. ACM, New York, NY, USA, pp. 1–2.

Döllner, J., Buchholz, H., Brodersen, F., Glander, T., Jutterschenke, S., Klimetschek, A., 2005. Smartbuildings - a concept for ad-hoc creation and refinement of 3d building models. In: Gröger, G., Kolbe, T. (Eds.), 1st International Workshop of 3D City Models. No. 49. EuroSDR, Bonn, Germany, pp. 5–10.

Döllner, J., Hagedorn, B., 2007. Integrating urban gis, cad, and bim data by service-based virtual 3d city-models. In: 26th Urban and Regional Data Management: UDMS 2007 Annual. Taylor & Francis Ltd, Stuttgart, Germany, pp. 157–170.

Döllner, J., Kolbe, T. H., Liecke, F., Sgouros, T., Teichmann, K., May 2006. The virtual 3d city model of berlin - managing, integrating, and communicating complex urban information. In: 25th Urban Data Management Symposium (UDMS). pp. 15–17, online proceedings (http://www.udms.net/).

Hall, M. M., Smart, P. D., Tomko, M., Jones, C. B., 2009. Report and software for generating location expressions for captions. Deliverable 6.6 TRIPOD Project, http://tripod.shef.ac.uk/ (Accessed 12 November 2010).

Hoegner, L., Kumke, H., Meng, L., Stilla, U., 2007. Visualization of building models and factual data integrated by citygml. In: The 5th International Symposium on Mobile Mapping Technology (MMT 2007). Moskau, Russia, pp. (On CD–ROM).

Jones, C. B., Ware, J. M., Miller, D. R., 1999. A probabilistic approach to environmental change detection with area-class map data. In: ISD '99: Selected Papers from the International Workshop on Integrated Spatial Databases, Digital Images and GIS. Lecture Notes in Computer Science Vol. 1737, Springer, Berlin, pp. 122–136.

Kolbe, T., Gröger, G., Plümer, L., March 2005. Citygml – interoperable access to 3d city models. In: Oosterom, Zlantanova, Fendel (Eds.), Proceedings of the 1st International Symposium on Geoinformation for Disaster Management. Delft, Springer, Verlag, pp. 21–23.

Kumke, H., 2003. Salzburg as 3d city model. Presented at Data Mining and Workflow, AGIT 2003, GISCluster, 2-4 June, Salzburg (Austria).

Kumke, H., Hoegner, L., Meng, L., Stilla, U., 2007. Visualization of building models and factual data integrated by citygml. In: Proceedings 5th International Symposium on Mobile Mapping Technology, MMT07. pp. (On CD–ROM).

Lee, S., Wang, Y., Lee, E., 2004. Compactness measure of digital shapes. In: Region 5 Conference: Annual Technical and Leadership Workshop. Norman, Oklahoma, USA. April, pp. 103–105.

Métral, C., Falquet, G., Cutting-Decelle, A., 2009. Towards semantically enriched 3d city models: An ontology-based approach. In: Kolbe, T., Zhang, H., Zlatanova, S. (Eds.), GeoWeb 2009 Academic Track - Cityscapes, International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences (ISPRS), Vancouver, BC, Canada. pp. (On CD–ROM).

Neubauer, S., Over, M., Schilling, A., Zipf, A., 2009. Virtual cities 2.0: Generating web-based 3d city models and landscapes based on free and user generated data (openstreetmap). In: GeoViz 2009. Contribution of Geovisualization to the concept of the Digital City. Workshop. Hamburg, Germany, pp. (On CD–ROM).

Quinn, J., Smart, P., Jones, C., 2009. 3d city registration and enrichment. In: In Proceedings of ISPRS COST Workshop on Quality, Scale and Analysis Aspects of City Models. Lund, Sweden, pp. (On CD–ROM).

Ross, L., Bolling, J., Döllner, J., Kleinschmit, B., 2009. Enhancing 3d city models with heterogeneous spatial. In: 12th AGILE International Conference on GI Science, Lecture Notes in Geoinformation and Cartography. Springer, pp. 113–133.

Samal, A., Sheth, S., Cueto, K., 1994. A feature-based approach to conflation of geospatial sources. International Journal of Geographical Information Science 18 (5), 459–489.

Schenk, S., Saathoff, C., Staab, S., Scherp, A., 2009. Semaplorer - interactive semantic exploration of data and media based on a federated cloud infrastructure. Journal of Web Semantics 7 (4), 298–304.

Smart, P., Quinn, J., Hall, M. M., Jones, C. B., Tomko, M., Dias, E., Fruijtier, S., van Exel, M., Pouwels, C., 2009. Report and prototype on use of 3d data for caption enhancement. Deliverable 6.4 TRIPOD Project, http://tripod.shef.ac.uk/ (Accessed 12 November 2010).

Stojmenović, M., Žunić, J., 2008. Measuring elongation from shape boundary. In: Journal of Mathematical Imaging and Vision. Vol. 30 (1). Springer Netherlands, pp. 73–85.

Walter, V., Fritsch, D., July 1999. Matching spatial data sets: A statistical approach. International Journal of Geographical Information Science 13 (5), 445–473.

Zadeh, L. A., 1965. Fuzzy sets. Information and Control 8, 338–353.