A Method for Matching Crowd-sourced and Authoritative Geospatial Data

Heshan Du, Natasha Alechina, Michael Jackson, Glen Hart
University of Nottingham, UK

Abstract

A method for matching crowd-sourced and authoritative geospatial data is presented. A level of tolerance is defined as an input parameter as some difference in the geometry representation of a spatial object is to be expected. The method generates matches between spatial objects using location information and lexical information, such as names and types, and verifies consistency of matches using reasoning in qualitative spatial logic and description logic. We test the method by matching geospatial data from OpenStreetMap and the national mapping agencies of Great Britain and France. We also analyze how the level of tolerance affects the precision and recall of matching results for the same geographic area using 12 different levels of tolerance within a range of 1 to 80 meters. The generated matches show potential in helping enrich and update geospatial data.

1. Introduction

Maps, whether digital or paper-based, are a common feature of our daily life. They typically provide a two-dimensional representation of geographic features, such as roads, rivers, buildings, places, etc., in the real world (i.e. a topographic base) over which other 'thematic' information may be displayed such as density of population or crime statistics. The information represented provides both an indication of where on the earth's surface an object of interest is (i.e. its geometry) and lexical information on what that geometry represents (e.g. a road and its name such as 'High Street'). Such information represented in maps is often referred to as geospatial data and plays an essential role in many governmental, economic and social operations, such as disaster response, urban planning and tourism.

Traditionally, most national level mapping was carried-out by government agencies or specialist mapping companies, because it required the use of expensive or difficult-to-obtain survey data, plus specialist tools and later software and an associated high-level of expertise. Geospatial data which is surveyed and classified using formal quality assurance procedures, for example by a national mapping agency, is referred to be 'authoritative'. Maps produced by the general public, who did not have access to such data sources, nor the specialist tools and software, focused more on smaller areas and on indicating where key features were in relative terms but typically could not be relied upon for precise location, completeness or consistency. This situation has been radically changed in recent years by a number of technological developments and by governments through the release of associated data (e.g. precise Global Navigation Satellite System data, satellite and aerial imagery). Perhaps the most important of these developments is the mobile smartphone. Such phones are capable of accurately recording their positions and combined with the use of simple-to-use applications can delimit physical and man-made features and tag the resulting geometries with information describing the nature, purpose and use of those features. This 'crowd-sourced data' may be actively collected as a volunteer activity by citizens (Goodchild 2007) or passively acquired as a bi-product of an application the main purpose of

which is something else. The concept of 'crowd-sourced geospatial data' was expressed in different ways, such as citizen science, volunteered or involuntary geospatial information, user-generated content, public participation or collaboratively contributed geographic information and neogeography, in literature from 1990 to 2014 (Goodchild 2007, Heipke 2010, Comber *et al.* 2014). OpenStreetMap (OSM) (OpenStreetMap 2014) is the most popular map project of crowd-sourced data. Compared to authoritative data, crowd-sourced data is usually less geometrically accurate, less formally structured and lacks the associated metadata that allows it to be used in situations where commercial, policy or life-critical use is involved (Jackson *et al.* 2010). However, crowd-sourced geospatial data still offers great potential as it often contains richer user-based information, can reflect real world changes (e.g. new constructions of buildings) more quickly, and has a much lower acquisition cost. It is desirable to use authoritative and crowd-sourced data to complement each other in order to provide a more complete, up-to-date, people-centric and richer picture of geospatial data. One promising application of this is to use crowd-sourced geospatial data to help national mapping agencies enrich and update authoritative data.

Governments invest large amounts of money in national mapping agencies, which act as the primary source of geospatial information in many countries. In order to provide the most up-to-date maps to customers, it is essential for national mapping agencies to update their data frequently and regularly. However, this is expensive in both time and money. Taking Ordnance Survey of Great Britain (OSGB) (Ordnance Survey 2014a), Great Britain's national mapping authority, as an example, according to its agency performance monitors, one of the OSGB 2013-2014 targets is 'some 99.6% of significant real-world features greater than six months old are represented in the database' (Ordnance Survey 2014b). To achieve this, OSGB employs a number of different methods:

- Major construction companies are contracted to provide change intelligence concerning where and when they will build and site plans enabling Ordnance Survey to schedule field survey in a timely fashion. This will capture a significant amount of change intelligence related to all major building sites, road construction and other large construction events.
- OSGB collects planning permissions from local authorities.
- OSGB receives change reports from individual surveyors who have observed any change in their local areas.
- OSGB captures further changes using aerial imagery. This can be used to capture missed major changes, such as a single house and a farm barn (that does not require a planning permission). It will also capture a lot of minor changes, such as new or removed hedgerows and paths.
- OSGB also receives change reports (e.g. letters, emails or phone calls) from the general public, but these reports only comprise a very small proportion of all the intelligence received.

For OSGB, minor changes are the most problematic, such as small buildings constructed by small private building companies, change of function (e.g. a country house is changed to a hotel), natural changes (e.g. a change of vegetation type or coastal erosion), extensions and alterations to buildings, private roads (either new built or modified). In summary, most major changes will be captured by OSGB, but there is a higher likelihood that small changes in buildings and changes to attributions (e.g. change of purpose) will be missed. Capturing this information is

becoming increasingly important as OSGB moves from being simply a map producer to one that wishes to supply much richer geographic information.

As shown by the example of OSGB, current working methods employed by national mapping agencies leave room for improvement and are faced with challenges raised by the rapid development of crowd-sourced geospatial data. As EuroGeographics' President, Ingrid Vanden Berghe (Geospatial PR 2014), says, 'Europe's National Mapping, Cadastral and Land Registry Authorities must adapt their activities to become geospatial information brokers if they are to continue to meet society's expectations'. This indicates that national mapping agencies will collate data rather than just collect data in future, except for areas where only national mapping agencies are able to collect the data.

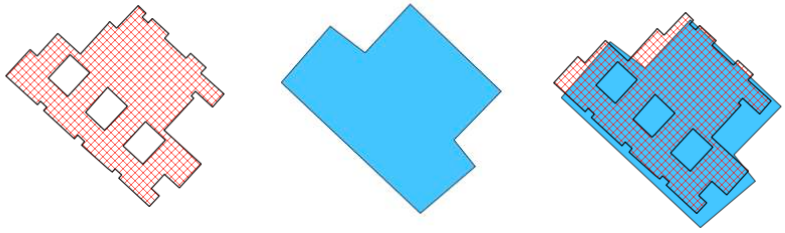

Figure 1: Huntingdon Primary and Nursery School represented in OSGB (stippled), OSM (solid) and their relative position

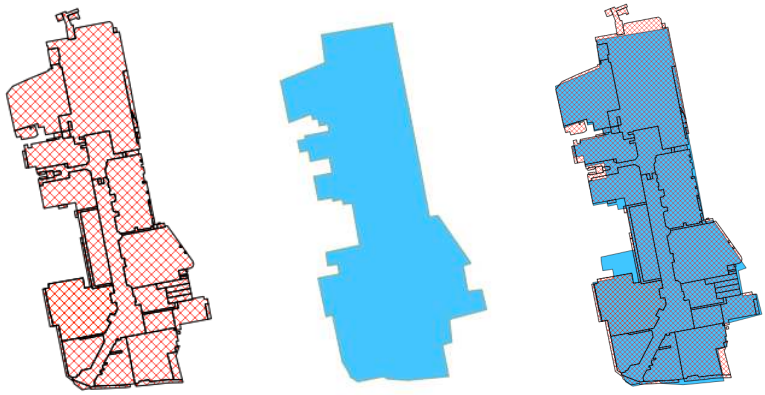

Figure 2: Victoria Shopping Centre represented in OSGB (stippled), OSM (solid) and their relative position

To use crowd-sourced data for enriching and updating authoritative data, it is essential to establish correspondence (matches) between spatial features represented in crowd-sourced and authoritative geospatial data. By using the established matches, descriptions about the corresponding spatial features can be identified and compared, which facilitates information validation, exchange and enrichment across datasets. In addition, using the established matches, spatial features which are not matched can be found. Descriptions about such unmatched spatial features in one dataset often contain additional useful information (e.g. more recent or up-to-date data) for the other dataset. Several real world application examples will be described later in this paper.

However, matching disparate geospatial data is far from straightforward. In different geospatial datasets, different terminologies or vocabularies are often used to describe spatial features. For example, the same restaurant may be classified as a *Restaurant* in one dataset, whilst as a *Place to Eat* in another database and simply as a brand-name in others (e.g. McDonald's). An identically spelt word, even within a single language, can often have many different meanings. Whilst an authoritative dataset will have a defined taxonomy or ontology where a word should have a precise definition, the 'crowd' may not follow such rules and may use several descriptions for a common object some of which may be local vernacular terms. For example, the word *College* may mean an institution within a university in one dataset, refer to a government secondary school in another and a private language training establishment in a third. Other terms may be used inconsistently, for instance, one person may include McDonald's within the category restaurant whilst others may not. For the same geographic area or the same set of spatial features, different geospatial data sources will have different representative geometries. Features may be represented in one dataset, but not in the other. The scale or accuracy of the geometry capture may vary. Even where the same precision of measurement is adopted, different points may be captured to represent the boundary of a feature so that two independently captured representations of a single object will always differ in some respect. As shown in Figure 1, the position and shape of Huntingdon Primary and Nursery School are represented differently in OSGB data (stippled) and OSM data (solid). In Figure 2, the Victoria Shopping Centre is represented as several shops in OSGB, but as a whole in OSM.

In this paper, we present a generic method for matching spatial objects held in different datasets with no shared form of digital identity. A *spatial object* in a geospatial dataset has an ID, location information and meaningful labels, such as names or types, and represents an object in the real world. A *geometry* here refers to a point, a line or a polygon, which is used to represent location information in geospatial datasets. We use both location information and lexical information to generate matches, and then check consistency of matches using reasoning in qualitative spatial logic and description logic. This idea has been implemented as a software tool called MatchMaps and its main steps have been described briefly in Du *et al.* (2015b), but without providing detailed algorithms for generating matches. This is what we do in this paper. To tolerate slight differences in geometric representations for the same spatial feature, the matching algorithms use a level of tolerance $\sigma \in R_{\geq 0}$ as input. The evaluation presented in Du *et al.* (2015b) is extended to show how the value of σ affects matching results. In Du *et al.* (2015b), the method was used to match OSM data and OSGB data. In this paper, we additionally use the method to match OSM data and data from IGN (Institut Géographique National 2014), the national mapping agency of France.

The rest of this paper is structured as follows. Section 2 reviews related work on geospatial data matching. Section 3 presents algorithms for matching geometries and spatial objects. Section 4 explains how the value of σ affects matching results and shows the generality of the method using IGN data. Section 5 discusses the practical use of the matches using real world examples. Section 6 concludes the paper and indicates possible future work.

2. Related Work

Geospatial data matching is defined as the task of identifying corresponding spatial features between different geospatial datasets. It is an essential step for data comparison, data integration or enrichment, change detection and data update. Over the last few decades, many methods (such as Walter and Fritsch 1999, Mustire and Devogele 2008, Tong et al. 2009, Safra et al. 2010, 2013, Li and Goodchild 2011, Huh et al. 2013, Tong et al. 2014) have been developed for matching authoritative geospatial data. Du (2015) provided a summary of these methods and discussed the limitations of them. None of these methods have been widely accepted and generally applied. The methods designed for matching authoritative geospatial data are not very suitable for OpenStreetMap (OSM) data, due to the information incompleteness and inaccuracy in OSM data, as well as its informal or non-standard representations. With the development of crowd-sourced geospatial data, several attempts have been made in order to match crowd-sourced geospatial data and authoritative geospatial data in the last few years.

Anand *et al.* (2010) applied map matching techniques to match road networks by calculating average distance and angle. However, it is computationally expensive and limited to linear features.

Ludwig *et al.* (2011) implemented an automated procedure for matching street networks of Navteq and OSM in Germany. Geometries and thematic attributes are compared to generate matches. However, it is specifically designed for business and geomarketing purpose, excluding features of no business interest.

Du *et al.* (2011) defined the meaning of 'same feature' regarding positional closeness, name similarity, category similarity and neighbourhood similarity. Then the probability of two spatial features being the same is calculated using a weighted function taking all these parameters into account. This work is preliminary and leaves the task of assigning weights of parameters to users.

Du *et al.* (2012) defined geometry consistency and topological consistency for road networks. Two lines are geometrically consistent with respect to a level of tolerance σ, if and only if they fall into the σ-buffer of each other. Topological consistency is checked using a description logic reasoner Pellet (Sirin *et al.* 2007), by comparing values of a functional data property 'neighbour set'. A neighbour set stores all the neighbours of an edge (two edges are neighbours if they have the same node). However, checking such topological consistency is too strict, due to inaccuracy and incompleteness of OSM data.

Koukoletsos *et al.* (2012) proposed an automated matching method for linear data in order to assess the completeness of OSM data compared to OSGB. It consists of seven stages and uses distance, orientation and attribute (road name and type) similarity constraints to generate and refine matches. However, with the existence of topological inconsistencies in OSM data, the method is not very efficient. In addition, the method does not handle abbreviations (which exist in OSM data) well when matching attributes.

Yang *et al.* (2013) proposed a heuristic probabilistic relaxation approach to match road networks. They use buffers to obtain candidate matches, then refine them by shape (dis)similarity (defined by distance, orientation and length) and structural similarity. The experimental results of

matching OSM and authoritative data are of high precision. However, the method is computationally expensive, and does not use attribute data, like road names.

Yang *et al.* (2014) proposed a method for matching points of interest from a crowd-sourced dataset and road networks from an authoritative dataset. It first constructs a connectivity graph by mining linear cluster patterns from points, then matches nodes in the graph to roads by probabilistic relaxation and a vector median filtering. The method assumes that linear patterns exist among the points. The performance of the method mainly depends on the clustering result of points.

Fan *et al.* (2014) introduced a method for matching building footprints (polygons), in order to assess the quality of OSM data. Their similarity measure is defined by the percentage of overlap area, using 30% as the threshold for matching footprints. By the experimental result of the study area in Munich, the method achieves very high precision and recall, both over 99%. However, the similarity measure will fail, for example, when the same building is represented as two disjoint polygons in OSM data and authoritative data.

Most of the methods discussed above are designed for matching roads or other linear features (except Fan *et al.* 2014) and do not support the verification of matches (except Du *et al.* 2012). In this paper, we present a new method for matching crowd-sourced and authoritative geospatial data. It uses both location information and lexical information such as names and types to generate matches, and verifies consistency of matches using reasoning in description logic and qualitative spatial logic. The method was used to match buildings and places (polygonal features) represented in several real world datasets. In experiments, it achieved high precision and recall, as well as reduced human effort.

3. Method

In this section, we present a method for matching spatial features in disparate geospatial datasets. The method consists of two main steps: matching geometries and matching spatial objects. The geometry matching is based on the concepts of 'possibly partOf' and 'possibly sameAs'. Section 3.1 explains algorithms used for matching geometries. Section 3.2 describes a procedure following which spatial objects are matched using geometry matches and lexical information. The method has a wider application than matching authoritative and crowd-sourced data and could be applied wherever it is necessary to match two geospatial datasets of vector data.

3.1 Matching Geometries

Since geometries in a crowd-sourced dataset may not be very accurate, when matching them to geometries in an authoritative dataset, a level of tolerance or margin of error is needed to tolerate slight differences in geometric representations for the same feature. With respect to a level of tolerance $\sigma \in R_{\geq 0}$, two new spatial relations *BPT* and *BEQ* are defined as follows and illustrated in Figure 3. They formalize 'possibly partOf' and 'possibly sameAs' respectively.

Figure 3: The three hatched red circles are buffered part of (BPT) the solid blue circle (left); Buffered Equal or BEQ (right)

Definition 3.1: According to ISO19107 (ISO Technical Committee 211 2003), the buffer of a geometry g is a geometry which contains exactly all the points within σ distance from g, where $\sigma \in R_{\geq 0}$. This is formalized as:

$$buffer(g,\sigma) = \{p \mid \exists\ q \in g : d(p,q) \in [0,\sigma]\}.$$

buffer(g, σ) and g are in the same reference system and dimension.

Definition 3.2: Let $\sigma \in R_{\geq 0}$ denote a level of tolerance. For two geometries g_1 and g_2, $BPT(g_1,g_2)$ (g_1 is buffered part of g_2), *iff* $g_1 \subseteq buffer(g_2,\sigma)$; $BEQ(g_1, g_2)$ (g_1 and g_2 are buffered equal), *iff* $BPT(g_1, g_2)$ and $BPT(g_2, g_1)$.

If *BEQ* and *BPT* are defined by an appropriate level of tolerance $\sigma \in R_{\geq 0}$, then for geometries X and Y, if $BEQ(X, Y)$, then X and Y possibly represent the same real world location, otherwise, they represent different locations. Similarly, if $BPT(X, Y)$, then X represents a location which is possibly part of what Y refers to. The geometry matching method presented in this section is based on this rationale, and takes a level of tolerance $\sigma \in R_{\geq 0}$ as input for matching two sets of geometries. This σ denotes the maximal difference between geometric representations of the same spatial features from input datasets. The value of σ can be established empirically by looking at two datasets side by side and matching geometries of features (e.g. landmarks) which are known to be the same.

The geometry matching method consists of two main algorithms, Algorithm **2** and Algorithm **3**, which generate *BPT* and *BEQ* matches respectively, by calculating and comparing the minimal σs (Definition 3.3).

Definition 3.3: A level of tolerance $\sigma \in R_{\geq 0}$ is minimal with respect to geometries g_1 and g_2, iff $g_1 \subseteq buffer(g_2,\sigma)$ holds and for any $\beta \in R_{\geq 0}$ and $\beta < \sigma$, $g_1 \subseteq buffer(g_2,\beta)$ does not hold. The minimal level of tolerance with respect to g_1 and g_2 is denoted as $min\sigma(g_1, g_2)$.

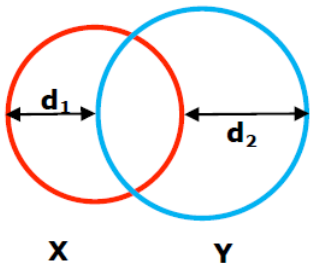

Figure 4: $min\sigma(X, Y) = d_1$ and $min\sigma(Y, X) = d_2$

The measure $min\sigma$ is not symmetric. As shown in Figure 4, X is a red circle and Y is a blue circle. Then the minimal level of tolerance with respect to X and Y is d_1, whilst the minimal level of tolerance with respect to Y and X is d_2. Though defined independently, the minimal level of tolerance was proved to be a measure equivalent to the directed Hausdorff distance, which is a generic measure for geometries (Du, 2015).

Algorithm **1** provides a way to calculate the minimal σ with respect to geometries g_1 and g_2 approximately. The input real numbers l and u denote a lower bound and an upper bound of $\sigma \in R_{\geq 0}$ respectively: $\sigma \in [l, u]$, $l \in R_{\geq 0}$, $u \in R_{\geq 0}$. The number $\beta \in R_{\geq 0}$ denotes the accuracy level, such that the absolute difference between the calculated value and the actual value of σ is no larger than β. Algorithm **1** does a 'binary search' between the lower bound l and the upper bound u of σ. It terminates and returns a calculated value m for the minimal σ, if m is accurate enough (Line 3) or a boundary case is reached, where $g_1 \subseteq buffer(g_2, m)$ and the boundaries of g_1 and $buffer(g_2, m)$ are connected (Line 8, g_1 and $buffer(g_2, m)$ are equal or g_1 is a tangential proper part of $buffer(g_2, m)$).

Algorithm 1:

```
1:    function minσ(g1, g2, l, u, β)
2:        m = (l + u)/2
3:        if (u - l) ≤ β then return m
4:        end if
5:        if g1 ⊆ buffer(g2, m) then
6:            b1 = boundary(g1)
7:            b2 = boundary(buffer(g2, m))
8:            if b1 ∩ b2 is not empty then return m
9:            end if
10:           return minσ(g1, g2, l, m, β)
11:       else
12:           return minσ(g1, g2, m, u, β)
13:       end if
14:   end function
```

Algorithm **2** takes two sets of geometries G_1, G_2 and a level of tolerance $\sigma \in R_{\geq 0}$ as input. For each geometry g_1 in G_1, it calculates the best candidate h in G_2, and add $BPT(g_1, h)$ to the set of output matches $M_{G1 \rightarrow G2}$, if such an h exists. The minimal σ is used as the criterion to select the best candidates (Definition 3.4).

Definition 3.4: For a geometry g, a set of geometries S, a level of tolerance $\sigma \in R_{\geq 0}$, the geometry $h_1 \in S$ is the best candidate for g, iff $min\sigma(g, h_1) < \sigma$, and for any $h \in S$, $min\sigma(g,h) \geq min\sigma(g,h_1)$.

Algorithm 2:

```
1:  function bpt-match(G1, G2, σ)
2:      MG1→G2 = {}
3:      for g1 ∈ G1 do
4:          h = null
5:          for g2 ∈ G2 do
6:              if minσ(g1, g2) < σ then
7:                  σ = minσ(g1, g2)
8:                  h = g2
9:              end if
10:         end for
11:         if h != null then
12:             add BPT(g1, h) to MG1→G2
13:         end if
14:     end for
15:     return MG1→G2
16: end function
```

Algorithm **3** calculates *BEQ* matches using *BPT* matches generated by Algorithm **2**. For every geometry $g_2 \in G_2$, Algorithm **3** matches it to a geometry G_s which is a union of geometries in G_1, such that g_2 and G_s are buffered equal, if such a G_s exists. This is done as follows. For every geometry $g_2 \in G_2$, we firstly obtain a set S containing every $g_1 \in G_1$ such that $BPT(g_1, g_2)$ is in $M_{G1 \rightarrow G2}$ (Lines 2-5). Since each geometry $g \in S$ is buffered part of g_2, their union G_s is buffered part of g_2. If g_2 is also buffered part of G_s (Line 7), then g_2 and G_s are buffered equal. Generating *BEQ* matches between g_2 and G_s directly may have some side effects or noise, especially when G_s consists of several disconnected parts (G_s is multiple, Line 8). Three examples are shown in Figure 5, where in each, the blue solid geometry is buffered equal to the union of several red stippled geometries. The extra red stippled geometries actually do not have any correspondences. The best candidates found for them using Algorithm 2 are the blue solid geometries, but the matches are wrong since the level of tolerance allowed is too large. Algorithm **4** is designed to refine G_s in such case, by calculating and comparing the minimal σs (Definition 3.3).

Algorithm 3:

```
1:  function beq-match(G1, G2, σ)
2:      MG1→G2=bpt-match(G1, G2, σ)
3:      Mbeq = {}
```

```
4:          for g2 ∈ G2 do
5:              S = {g1 ∈ G1 | BPT(g1, g2) ∈ MG1→G2}
6:              Gs = ⋃g∈S g
7:              if BPT(g2, Gs) then
8:                  if Gs is multiple then
9:                      Gs = refine(Gs, g2, σ)
10:                 end if
11:                 add BEQ(g2, Gs) to Mbeq
12:             end if
13:         end for
14:         return Mbeq
15:     end function
```

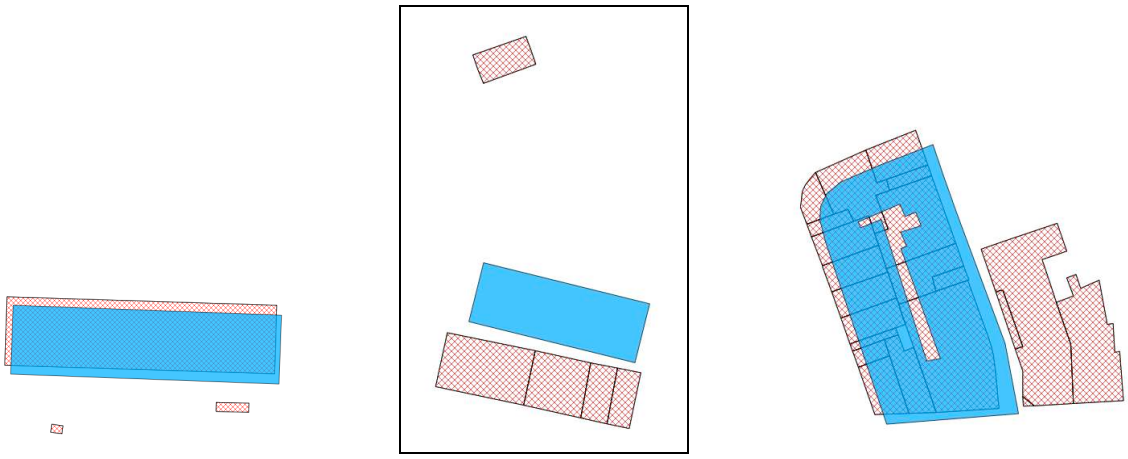

Figure 5: *BEQ* matches with 'noise'

Algorithm 4:

```
1:      function refine(Gs, g2, σ)
2:          s = minσ(g2, Gs)
3:          for g ∈ Gs.getGeometries() do
4:              if g2 contains g then continue
5:              end if
6:              remain = Gs \ g
7:              if BPT(g2, remain) does not hold then continue
8:              end if
9:              sr = minσ(g2, remain) // sr ≥ s
10:             if s = sr then return refine(remain, g2, σ)
11:             end if
12:             t = minσ(Gs, g2)
13:             tr = minσ(remain, g2)
14:             if (s + t) ≥ (sr + tr) then return refine(remain, g2, σ)
15:             end if
16:         end for
```

17: **return** G_s
18: **end function**

Algorithm **4** takes two geometries G_s, g_2 as input, where G_s is multiple and g_2 is not. G_s and g_2 are buffered equal with respect to the level of tolerance σ. Algorithm **4** refines G_s to a subset of it, and maintains the buffered equal relation as an invariant during the refining process. This is done as follows. For every geometry g contained in G_s, if g is not fully covered by g_2, then we obtain *remain*, which is G_s without g (Line 6). To maintain the invariant, we check whether *BEQ*(*remain,* g_2) holds. Since *BPT*(G_s, g_2) and *remain* $\subset C_s$, *BPT*(*remain,* g_2) already holds. Thus, we only need to check whether g_2 is buffered part of *remain*. If yes, the next steps in the for-loop are followed. We calculate the minimal σ (Definition 3.3) with respect to g_2 and G_s (Line 2), g_2 and *remain* (Line 9) as s and s_r respectively. By Definition 3.3, Definition 3.1 and *remain* $\subset G_s$, $s_r \geq s$. If s and s_r are equal, then we can remove g from G_s without changing the required buffer size (Line 10). After applying this, the extra red geometries in Figure 5 (left and middle) are removed, as shown in Figure 6 (left and middle) respectively. However, the extra geometries in Figure 5 (right) cannot be removed, because the boundary of the blue geometry is close to the red geometries outside, the existence of which makes the required buffer size smaller. For such case, we calculate the minimal σ with respect to G_s and g_2 (Line 12), *remain* and g_2 (Line 13), as t and t_r respectively. If $(s + t) \geq (s_r + t_r)$, we can remove g from G_s without making the sum of required buffer sizes larger (Line 14). Applying this removes the extra geometries in Figure 5 (right), as shown in Figure 6 (right). Algorithm 4 recursively removes one part from C_s and returns the remaining parts, until no parts can be removed.

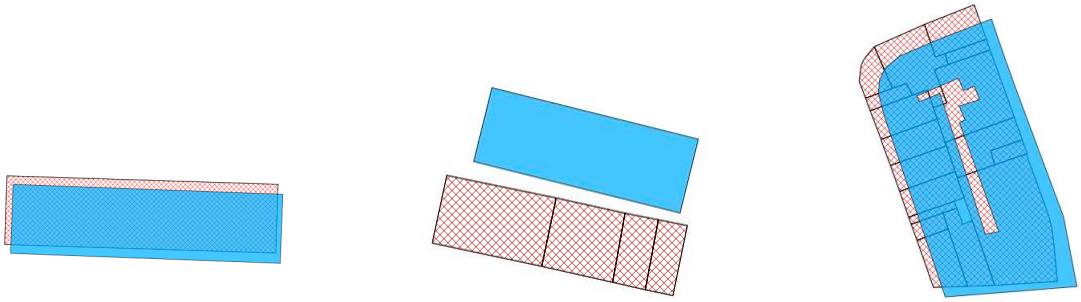

Figure 6: Refined *BEQ* matches

After applying Algorithm 4, Algorithm 3 generates and adds refined *BEQ* matches to its output mapping M_{beq}.

3.2 Matching Spatial Objects

In this section, we describe a method for matching spatial objects, making use of *BEQ* matches generated by Algorithm **3** and lexical descriptions (names and types) of spatial objects. A *sameAs* match between spatial objects a and b states that a and b represent the same real world object. This is denoted as *sameAs*(a, b). A *partOf* match from a spatial object a to a spatial object b states that a represents a real world object which is part of what b refers to. This is denoted as

partOf (*a, b*). The output of the object matching method is a set of *sameAs* and *partOf* matches between spatial objects. The method does not directly use *BPT* matches generated by Algorithm **2**, mainly because spatial objects and their parts may not have any similar lexical information.

As a function, *objects*(*g*) maps every geometry *g* to a set of spatial objects, where the geometry of each object $g_i \subseteq g$. For any pair of geometries g_1 and g_2 which are *BEQ*-matched, we match *objects*(g_1) and *objects*(g_2) based on the similarity of lexical information (names and types represented by strings).

The similarity measure for lexical information is described as follows. For strings s_1 and s_2, *similar*(s_1, s_2) is true, if s_1, s_2 are equal, one contains the other, one is an abbreviation of the other, or their Levenshtein edit distance is smaller than *length*(s_1)/2 or *length*(s_2)/2. For any spatial object *o*, let *names*(*o*) denote its set of names, *types*(*o*) denote its set of types. For any pair of spatial objects o_1,o_2, *similarNames*(o_1, o_2) is true, if there exist $n_1 \in$ *names*(o_1) and $n_2 \in$ *names*(o_2) such that *similar*(n_1, n_2). Otherwise, *similarNames*(o_1, o_2) is false. *similarTypes*(o_1, o_2) is defined in the same way as defining *similarNames*(o_1, o_2). For the type similarity, using string comparison is not sufficient, and more sophisticated similarity measures should be used to recognize different words expressing the same type, for example, *house*, *dwelling* and *residential*. Currently, such information is only hard-coded for houses. For spatial object *o*, *house*(*o*) is true, if the type of *o* is *house*, *dwelling* or *residential*. Otherwise, *house*(*o*) is false.

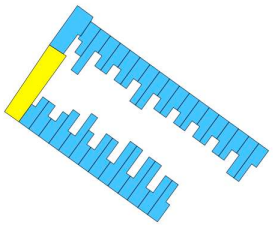

Figure 7: All are houses except one.

For any pair of geometries g_1 and g_2 which are *BEQ*-matched by Algorithm **3**, *objects*(g_1) and *objects*(g_2) are matched as follows:

Case 1: If $|objects(g_i)| = 0$, $i \in \{1, 2\}$, then there are no objects to match.

Case 2: If $|objects(g_i)| = 0$, $|objects(g_j)| > 0$, $i \mathrel{!}= j$, then objects in *objects*(g_j) do not have any corresponding objects.

Case 3: If $|objects(g_i)| = 1$, $i \in \{1, 2\}$, $o_i \in objects(g_i)$, *similarNames*(o_1,o_2) is true, or *names*(o_1) is empty, or *names*(o_2) is empty, then we generate a *sameAs* match between o_1 and o_2.

Case 4: If $|objects(g_i)| = 1$, $|objects(g_j)| > 1$, $i \mathrel{!}= j$, then:

a) If there exists exactly one object $o_j \in objects(g_j)$, such that for $o_i \in objects(g_i)$, *similarName*(o_i, o_j) is true, then we generate a *sameAs* match between o_i and o_j.
b) Otherwise, for each object $o_j \in objects(g_j)$, we generate a *partOf* match from o_j to $o_i \in$ *objects*(g_i).

Case 5: If $|objects(g_i)| > 1$, $i \in \{1, 2\}$, then:

a) If there exists at most one object o in $objects(g_i)$ such that $house(o)$ is false, and for any other object o_i in $objects(g_i)$, $house(o_i)$ is true, then we create an abstract object O_i corresponding to the aggregation of all objects in $objects(g_i)$. For every object $o_j \in objects(g_j)$, $i != j$, we generate a *partOf* match from o_j to O_i.

   As shown in Figure 7, there is only one spatial object (yellow) which is not a house, and all others are houses. Matching every spatial object is not interesting but requires much more effort than creating and matching an abstract object for them.

b) If no abstract object is created, then we match objects by their names first and then by their types.
   i. For objects $o_1 \in objects(g_1)$, $o_2 \in objects(g_2)$, if $similarNames(o_1, o_2)$, then we generate all possible matches: a *sameAs* match between o_1 and o_2, *partOf* matches from o_1 to o_2 and from o_2 to o_1.
   ii. For 'not-matched' objects $o_1 \in objects(g_1)$, $o_2 \in objects(g_2)$, if at least one of $names(o_1)$ and $names(o_2)$ is empty, and at least one of $similarTypes(o_1, o_2)$ and $(house(o_1) \wedge house(o_2))$ is true, then we generate a *sameAs* match between o_1 and o_2, *partOf* matches from o_1 to o_2 and from o_2 to o_1.

   Then we use our new qualitative spatial logic LBPT (Du and Alechina 2014a, b) to verify consistency of the generated matches with respect to location information. LBPT was designed for reasoning about geometries represented in different geospatial datasets, in particular crowd-sourced datasets. The relations between geometries considered in the logic are: BPT, Near and Far. By Definition 3.2, BEQ is definable by BPT. The relations Near and Far are also defined using a level of tolerance σ. LBPT formalizes different cases where a contradiction can be detected by LBPT reasoning. For any pair of spatial objects a and b, we assumed that if $sameAs(a, b)$ is true, then the geometry of a and the geometry of b are BEQ; if $partOf(a, b)$ is true, then the geometry of a is BPT the geometry of b, where BEQ and BPT are defined using an appropriate level of tolerance σ. A contradiction exists, for example, if spatial objects a_1 is *sameAs* a_2, b_1 is *sameAs* b_2, a_1 and b_1 are near, but a_2 and b_2 are far. As an optional step, we could use description logic to verify consistency of the generated matches with respect to UNA/NPH (Unique Name Assumption/No PartOf Hierarchy) (Du et al. 2015b) after using spatial logic. For example, an inconsistency exists, if a spatial object is stated as being *sameAs* two spatial objects in another dataset. This step could be skipped if UNA/NPH is violated frequently in an input dataset.

Finally, we use description logic to verify consistency of all generated matches with respect to classification information. For example, an inconsistency exists, if $sameAs(o_1, o_2)$, o_1 is a *Bank*, o_2 is a *Clinic*, *Bank* and *Clinic* are disjoint, containing no common element. If any inconsistency is detected by reasoning in spatial logic or description logic, minimal sets of statements for deriving it will be generated and visualized to a domain expert for deciding which statement is wrong and should be retracted to restore consistency. The detailed explanations for using spatial logic and description logic to verify matches have been provided in (Du et al. 2013, 2015b).

4. Evaluation

In (Du *et al.* 2015b), we established the precision and recall of the method for matching OSM data (building layer) (OpenStreetMap 2014) and OSGB MasterMap data (Address Layer and Topology Layer) (Ordnance Survey 2014a) using a single value of σ (the level of tolerance). In (Du *et al.* 2015a), we evaluated the method with respect to the amount of human effort required for resolving contradictions detected by reasoning in spatial logic and description logic, and showed that the human effort was reduced compared to a fully manual matching process. In this section, we extend the previous evaluation in two ways. Firstly, we use several different values of σ for matching the same area represented in OSM data and OSGB MasterMap data and explain how the value of σ affects the matching results. The study area is in the city centre of Nottingham, UK. Secondly, we apply the method to match OSM data and IGN data (BD TOPO database, buildings and toponymy layers) (Institut Géographique National 2014) to show the generality of the method. The study area is a central area in Paris, France.

As explained already in (Du *et al.* 2015b), before applying the method presented in Section 3, we used standard 2D spatial tools to aggregate adjacent OSM geometries automatically so that a block of houses can be matched together. Figure 8 shows the geometric representations of the same block of houses in OSGB data (stippled) and OSM data (solid). From OSM data, we only know all of them are houses. Matching the houses one by one is time-consuming and not helpful for enriching OSGB/IGN data.

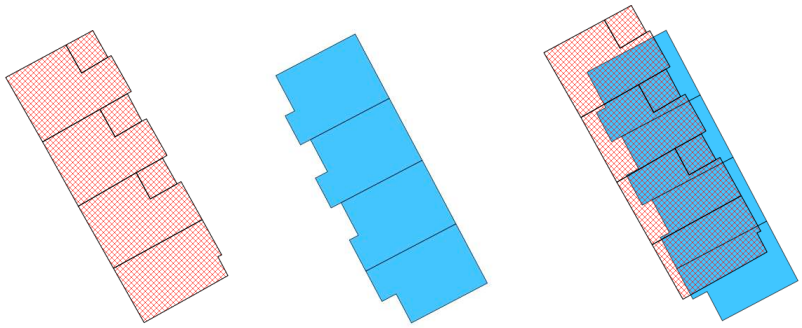

Figure 8: The geometric representations of the same block of houses in OSGB data (stippled), OSM data (solid) and their relative position.

4.1 Nottingham Case Study

The data used in the Nottingham case was obtained in 2012 and is shown in Figure 9. Its statistics are summarized in Table 1. The spatial objects in OSGB are generated using names of buildings and premises within buildings. The number of OSM spatial objects is smaller than that of OSGB, because OSM data often describes a collection of OSGB spatial objects as a whole, for example, many OSGB shops as one large shopping centre in OSM data.

Figure 9: The geometric representations of Nottingham city centre from OSGB (left) and OSM (right)

We apply the method several times to match spatial objects in the Nottingham case using a variety of σ values. The ground truth is established in the same way as explained in (Du *et al.* 2015b). For each OSM spatial object, we classify it into one of the following categories by checking all the generated matches involving it: 'Correctly Matched' (True Positive or *TP*), 'Incorrectly Matched' (False Positive or *FP),* 'Correctly Not-matched' (True Negative or *TN)* and 'Incorrectly Not-matched' (False Negative or *FN)*. If a spatial object is incorrectly matched but should be matched (i.e. there exists a correct match for it), then we label it as FP_{sbm}. Note that the size of each category is the number of OSM spatial objects in it. For example, for the Victoria Centre in OSM data, though there are hundreds of *partOf* matches involving it, it is only counted as one element in 'Correctly Matched'. Precision is computed as the ratio of $|TP|$ to $|TP| + |FP|$, and recall as the ratio of $|TP|$ to $|TP| + |FN| + |FP_{sbm}|$.

Table 2 summarizes the matching results for matching spatial objects in the Nottingham case using 12 different values of σ. From 0 to 80 meters, we take a value for every 10 meters. We also

take some other values (1, 3, 5 and 15 meters) because the precision/recall changes more quickly from 0 to 10 meters and from 10 to 20 meters. For the matching results obtained by taking 5 meters, 10 meters, 20 meters, 30 meters, 40 meters and 60 meters as the level of tolerance, Figure 10 visualizes the geometries of spatial objects in different categories as maps. In (Du *et al.* 2015b), we estimated the appropriate level of tolerance for the Nottingham case to be 20 meters and established the precision and recall. The matching results obtained here using σ =20 meters are slightly different from those presented in (Du *et al.* 2015b), because the graphical user interface was modified (see Du *et al.* 2015a) with simpler but fewer options provided to users for retracting wrong matches. Based on the matching results obtained using different σ values presented in Table 2, using σ = 20 meters achieves both relatively high precision and recall compared to others. This justifies that σ = 20 meters is appropriate and a good estimate. However, it is not the optimal, as the matching results obtained using σ = 30 meters are of the same precision but slightly higher recall than those obtained using σ = 20 meters. In the following, we provide a more detailed analysis on how the level of tolerance affects the precision and recall.

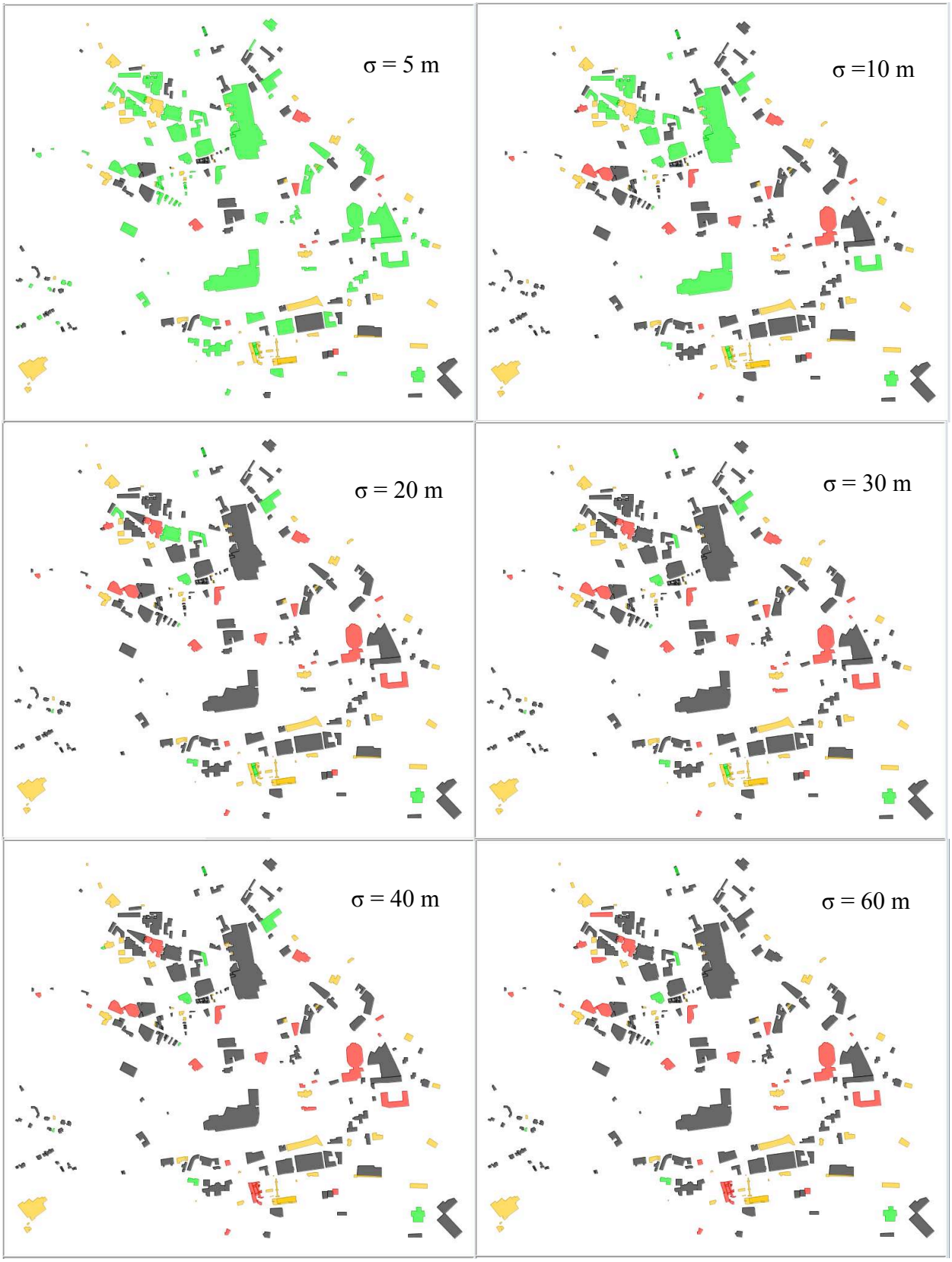

Figure 10: OSM spatial objects of the Nottingham case are classified into four categories: TP (Black), FP (Red), TN (Yellow) and FN (Green).

As shown in Table 2, when the level of tolerance σ is 1 meter, only 1 spatial object is correctly matched, and all others are not matched. Hence the recall is nearly 0. With the increase of the σ value, as shown in Figure 10, more spatial objects are correctly matched. For example, in Figure 11, the Arkwright building of the Nottingham Trent University is represented as a concave geometry in OSGB data but as a convex geometry in OSM data, which can be matched using σ = 30 meters but not σ = 20 meters.

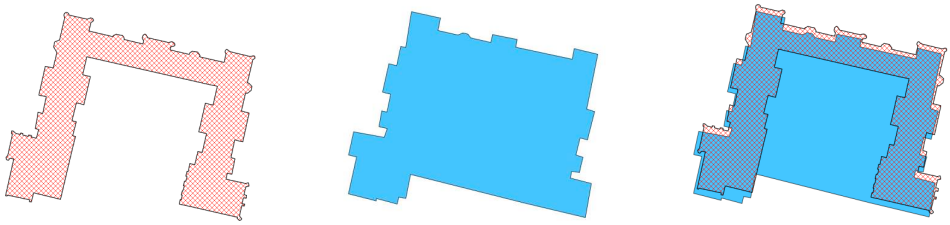

Figure 11: Nottingham Trent University's Arkwright building represented in OSGB (stippled), OSM (solid) and their relative position

As the increase of σ makes more spatial objects to be correctly matched, the recall increases rapidly to 0.78 when σ = 10 meters, as shown by Figure 12. Then the recall increases more and more slowly because fewer and fewer spatial objects can be correctly matched. It reaches 0.85 when σ =30 meters, and stays the same until σ = 80 meters. When σ = 80 meters, there are still spatial objects which are incorrectly not-matched. The method cannot match them, mainly because the lexical matching used by the method cannot match different names (represented by non-similar strings) of the same real world object. For example, the OSGB spatial object labeled as `Nottinghamshire Constabulary, Police Services' and the OSM spatial object labeled as `Central Police Station' cannot be matched but actually represent the same object in the real world.

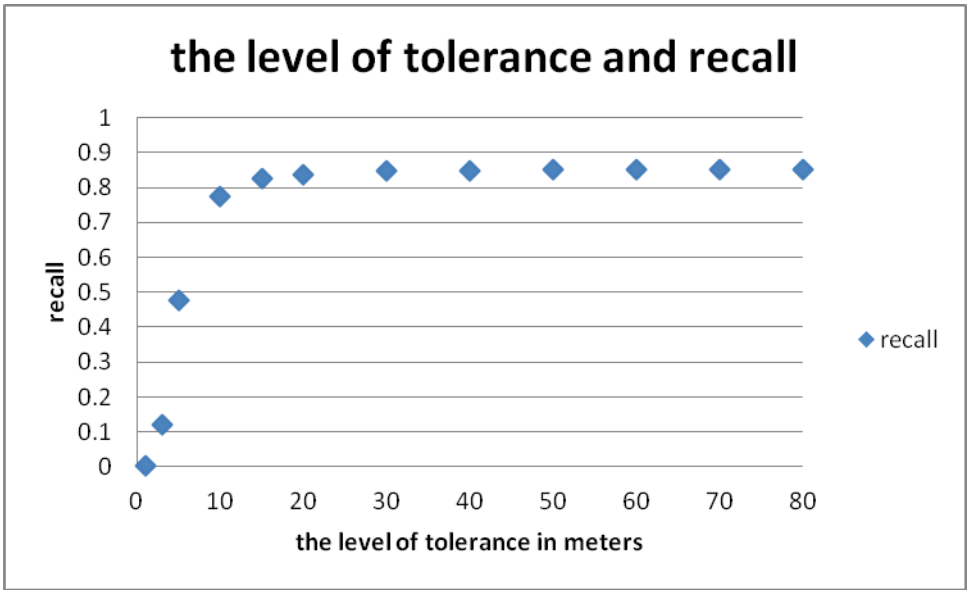

Figure 12: the level of tolerance and recall

As shown in Table 2 and Figure 13, increasing the level of tolerance σ from 1 to 80 meters, the precision falls but it is always ≥ 0.8. The precision becomes lower when σ increases, mainly because a larger level of tolerance makes more spatial objects to be incorrectly stated as being *partOf* some other spatial objects nearby. It is difficult to prevent such mistakes because spatial objects and their parts may not have any similar lexical information and therefore *partOf* matches are generated mostly based on geometry matching. Though the generated matches will be verified using reasoning in spatial logic and description logic, not all mistakes can be detected. For example, increasing the value of σ from 30 to 40 meters, the main concourse, ticket office, travel center and some other offices or shops within the Nottingham train station represented in OSM are all incorrectly stated as being *partOf* the Xpress Catering within the Nottingham trains station represented in OSGB, as their geometries are matched. Such wrong *partOf* matches are not detected by spatial logic because the objects involved are all near to each other. They are not detected by description logic because some OSM spatial objects do not have any type information and the use of description logic for verifying consistency of *partOf* matches (Du *et al.* 2015b) is limited by a small set of manually generated `partOf-disjointness' statements (e.g. a *School* cannot be *partOf* a *Pub*) and does not cover the types involved in the wrong matches. As a result, the precision drops from 0.89 to 0.83. Despite this, the precision is quite stable when σ varies from 5 to 30 meters and from 40 to 80 meters, staying around 0.9 and 0.82 respectively.

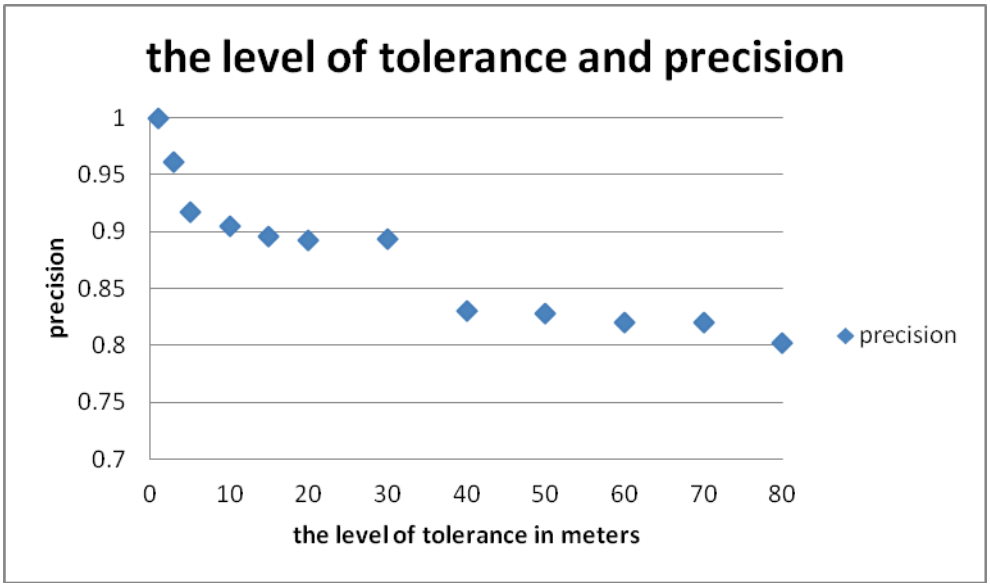

Figure 13: the level of tolerance and precision

In summary, based on the Nottingham case study, the performance of the method is quite stable using a level of tolerance from 15 to 80 meters. The precision of matching results is within a range of 0.8 to 0.9. The recall is within a range of 0.83 to 0.85.

4.2 Paris Case Study

In this section, we report the use of the method for matching OSM data and IGN data (BD TOPO database, buildings and toponymy layers) (Institut Géographique National 2014). The study area is a central area in Paris, France. The data used in the Paris case was obtained in 2013 and is shown in Figure 14. Its statistics are summarized in Table 3. Differing from OSGB MasterMap data, the IGN BD TOPO database does not contain any names of premises within buildings. Therefore, the spatial objects in IGN are generated only using names of buildings. Since most of the buildings in IGN data do not have a name, the number of spatial objects in IGN data is small.

Figure 14: The geometric representations of the central area of Paris from IGN (left) and OSM (right)

We set the value of σ to be 40 meters such that the Île de la Cité island in Paris can be matched. Interestingly, the positional accuracy of OSM data has been estimated to be about 40 meters in France (Girres and Touya 2010). The ground truth is established manually in the same way as explained for the Nottingham case. The geometries of OSM spatial objects which are 'Correctly Matched' (True Positive or TP), 'Incorrectly Matched' (False Positive or *FP*), 'Correctly Not-matched' (True Negative or *TN*) and 'Incorrectly Not-matched' (False Negative or *FN*) are visualized in Figure 15. Their statistics are shown in Table 4. The precision and recall are both ≥ 83%. Since the number of generated matches in the Paris case is small, the precision and recall are achieved by the method fully automatically. In other words, the reasoning in spatial logic and description logic does not detect any inconsistency and thus requires no human effort for retracting problematic matches. Whilst most of OSM spatial objects in the Nottingham case are correctly matched, most of those in the Paris case are correctly not-matched. This indicates OSM data contains much richer lexical information about names and types, which does not exist in the IGN BD TOPO database. Based on the Paris case study, the method is also applicable and effective for matching OSM data and IGN data.

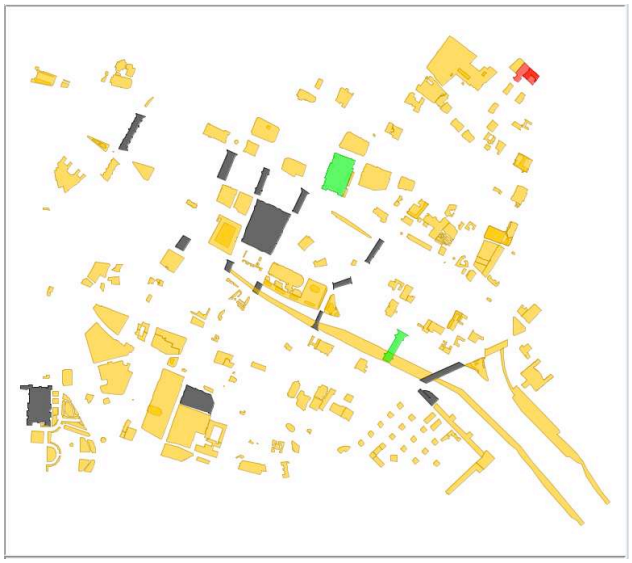

Figure 15: OSM spatial objects of the Paris case are classified into four categories: TP (Black), FP (Red), TN (Yellow) and FN (Green).

5. Application

The matches generated by the method have several practical uses. Firstly, the matches can help validate the correctness of corresponding data in input datasets. If similar records of a spatial feature exist in both input datasets which are developed independently, then the records have a higher chance of being correct. In addition, the matches facilitate information exchange and enrichment, as one dataset may contain more detailed lexical descriptions or more user-based information than the other. For example, classification descriptions of spatial features in OSM data can be more precise and more understandable by non-specialists. There are several spatial features in OSM data, such as shopping centres, hospitals and schools, which correspond to collections or aggregations of spatial features in OSGB.

Using the matches, spatial features which are not matched can be found. The 'not matched' spatial features in one dataset contain information which does not exist in the other dataset. For example, in the Paris case study, most OSM spatial objects are not matched and can be used to help enrich lexical information in the IGN BD TOPO database. Figures 16-18 show three examples in the Nottingham case, where spatial objects are not matched but their geometries are matched. By going to the real places to check, we found that at the location shown in Figure 16, there is a shop called 'New York Nails' but no 'Las Vegas Nails', so OSGB data is wrong and out of date. At the location shown in Figure 17, both Network Rail Ltd. and the NEMS Platform One Medical Practice exist and they are next-door. The NEMS Platform One Medical Practice is new and has not been reflected in OSGB data. The location shown in Figure 18 is called by people working there 'Eastcroft Incinerator'. It is operated by Wastenot Reclamation Ltd. whose location is somewhere else. In this case, OSM data and OSGB data describe different aspects of

the same location, and the OSM data provides more user-based information which can help enrich OSGB data.

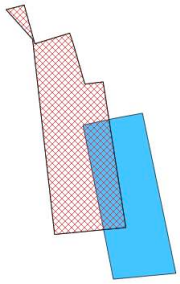

Figure 16: The geometries of the Las Vegas Nails in OSGB data (stippled) and the New York Nails in OSM data (solid) are BEQ-matched.

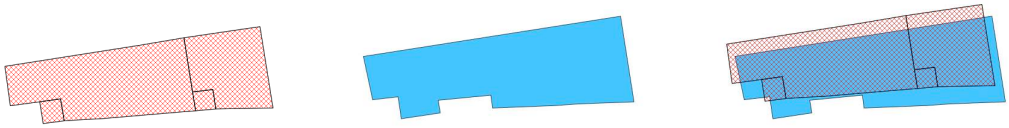

Figure 17: The geometries of the Network Rail Ltd. in OSGB data (stippled) and the NEMS Platform One Medical Practice in OSM data (solid) are BEQ-matched.

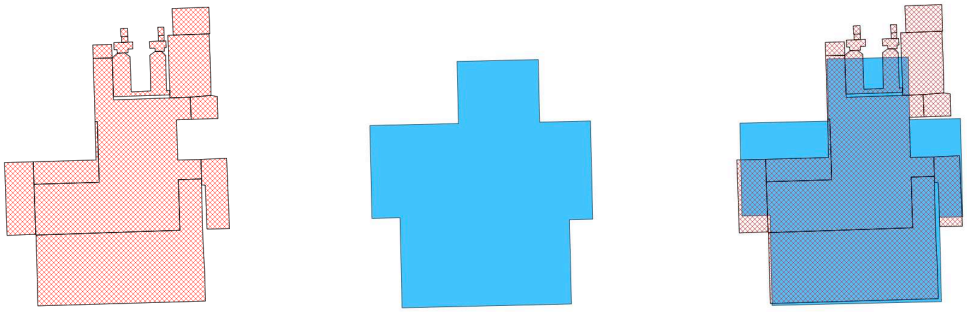

Figure 18: The geometries of the Wastenot Reclamation Ltd. in OSGB data (stippled) and the Eastcroft Incinerator in OSM data (solid) are BEQ-matched.

As explained in Section 1, OSGB collects information about real world changes from a variety of sources, such as major construction companies, local authorities, individual surveyors, aerial imagery, as well as reports from the general public. As illustrated by the examples above, the unmatched OSM spatial features have the potential to comprise a complementary source of change intelligence. This OSM change intelligence may not be as accurate as the others, but it is

free and can capture not only major changes but also many minor changes in buildings and roads noticed by OSM contributors, as well as changes in function or purpose. It is difficult for OSGB to capture such minor changes and functional changes using current methods. Using the OSM change intelligence seems promising but needs more advanced techniques for validating the correctness of crowd-sourced data and to be tested in practice.

6. Conclusions

In this paper, we present a generic method for matching crowd-sourced and authoritative geospatial data. It generates *sameAs* and *partOf* matches between spatial objects using both location and lexical information, and verifies consistency of matches using reasoning in qualitative spatial logic and description logic. The method is applied for matching OSM data, OSGB data and IGN data. For the Nottingham case, increasing the level of tolerance from 1 to 80 meters, the precision falls slowly and is always ≥ 0.8, the recall increases and converges at 0.85. For the Paris case, using 40 meters as the level of tolerance, a precision of 0.88 and a recall of 0.83 are achieved. Theoretically, the method presented can be used to match objects having polygonal, linear or point geometries. As future work, the generality of this method will be tested further by matching point or linear spatial features. In addition, we will use matches for enriching and updating geospatial data, and minimize the amount of human effort required during this process.

Acknowledgements

We express thanks to Ordnance Survey of Great Britain and Institut Géographique National of France for providing the test data.

References

Anand S, Morley J, Jiang W, Du H, Jackson M J, and Hart G 2010 When Worlds Collide: Combining Ordnance Survey and Open-StreetMap data. In *Proceedings of Association for Geographic Information (AGI) GeoCommunity '10 Conference*

Comber A, Schade S, See L, Mooney P, and Foody G 2014 Semantic analysis of Citizen Sensing, Crowdsourcing and VGI. In *Proceedings of the 17th AGILE International Conference on Geographic Information Science* ISBN: 978-90-816960-4-3

Du H 2015 *Matching Disparate Geospatial Datasets and Validating Matches using Spatial Logic*. PhD thesis, School of Computer Science, University of Nottingham.

Du H and Alechina N 2014a A Logic of Part and Whole for Buffered Geometries. In *Proceedings of the 21st European Conference on Artificial Intelligence (ECAI)*, 997–998

Du H and Alechina N 2014b A Logic of Part and Whole for Buffered Geometries. In *Proceedings of the 7th European Starting AI Researcher Symposium (STAIRS),* 91–100

Du H, Alechina N, Hart G, and Jackson M J 2015a A Tool for Matching Crowd-sourced and Authoritative Geospatial Data. In *Proceedings of the International Conference on Military Communications and Information Systems* (accepted, available at http://www.cs.nott.ac.uk/~hxd/paper/A-Tool-Du-ID31.pdf)

Du H, Alechina N, Jackson M, and Hart G 2013 Matching Formal and Informal Geospatial Ontologies. Lecture Notes in Geoinformation and Cartography *Geographic Information Science at the Heart of Europe.* Springer, 155–171

Du H, Anand S, Alechina N, Morley J G, Hart G, Leibovici D G, Jackson M J, and Ware J M 2012 Geospatial Information Integration for Authoritative and Crowd Sourced Road Vector Data. *Transactions in GIS,* 16 (4), 455–476

Du H, Jiang W, Anand S, Morley J, Hart G, and Jackson M J 2011 An Ontology-based Approach for Geospatial Data Integration. In *Proceedings of the 25th International Cartography Conference, CO*-118

Du H, Nguyen H H, Alechina N, Logan B, Jackson M J, and Goodwin J 2015b Using Qualitative Spatial Logic for Validating Crowd-Sourced Geospatial Data. In *Proceedings of the 29th AAAI Conference on Artificial Intelligence,* 3948-3953

ESRI 2014 ArcMap 10.1. Environmental Systems Resource Institute, Redlands, California

Fan H, Zipf A, Fu Q, and Neis P 2014 Quality assessment for building footprints data on OpenStreetMap. *International Journal of Geographical Information Science,* 28 (4), 700–719

Geospatial PR 2014 National Mapping, Cadastral and Land Registry Authorities look to future role as geospatial brokers. WWW document, http://geospatialpr.com/2014/10/14

Girres J F and Touya G 2010 Quality Assessment of the French OpenStreetMap Dataset. *Transactions in GIS,* 14 (4), 435–459

Goodchild M F 2007 Citizens as sensors: the world of volunteered geography. *Geo-Journal,* 69 (4), 211–221

Hart G, Dolbear C, Kovacs K, and Guy A 2008 Ordnance Survey Ontologies. WWW document, http://www.ordnancesurvey.co.uk/oswebsite/ontology

Heipke C 2010 Crowdsourcing geospatial data. *Journal of Photogrammetry and Remote Sensing,* 65 (6), 550 – 557

Huh Y, Yang S, Ga C, Yu K, and Shi W 2013 Line segment confidence region-based string matching method for map conflation. *Journal of Photogrammetry and Remote Sensing,* 78 (0), 69–84

ISO Technical Committe 211 2003 *ISO 19107:2003 Geographic information – Spatial schema.* Technical report, International Organization for Standardization (TC 211)

Jackson M J, Rahemtulla H, and Morley J 2010 The Synergistic Use of Authenticated and Crowd-Sourced Data for Emergency Response. In *Proceedings of the 2nd International Workshop on Validation of Geo-Information Products for Crisis Management,* 91–99

Koukoletsos T, Haklay M, and Ellul C 2012 Assessing Data Completeness of VGI through an Automated Matching Procedure for Linear Data. *Transactions in GIS,* 16 (4), 477–498

Li L and Goodchild M F 2011 An optimisation model for linear feature matching in geographical data conflation. *International Journal of Image and Data Fusion,* 2 (4), 309–328

Ludwig I, Voss A, and Krause-Traudes M 2011 A Comparison of the Street Networks of Navteq and OSM in Germany. Lecture Notes in Geoinformation and Cartography *Advancing Geoinformation Science for a Changing World.* Springer, 65–84

Mustire S and Devogele T 2008 Matching Networks with Different Levels of Detail. *GeoInformatica,* 12 (4), 435–453

OpenStreetMap 2014 WWW document, http://www.openstreetmap.org

Ordnance Survey 2014a WWW document, http://www.ordnancesurvey.co.uk

Ordnance Survey 2014b Agency performance monitors — Ordnance Survey's performance targets. WWW document, http://www.ordnancesurvey.co.uk/about/governance/agency-performance-monitors.html

Safra E, Kanza Y, Sagiv Y, and Doytsher Y 2006 Efficient Integration of Road Maps. In *Proceedings of the 14th Annual ACM International Symposium on Advances in Geographic Information Systems*, 59–66

Safra E, Kanza Y, Sagiv Y, Beeri C, and Doytsher Y 2010 Location-based algorithms for finding sets of corresponding objects over several geo-spatial data sets. *International Journal of Geographical Information Science,* 24 (1), 69–106

Safra E, Kanza Y, Sagiv Y, and Doytsher Y 2013 *Ad hoc* matching of vectorial road networks. *International Journal of Geographical Information Science,* 27 (1), 114–153

Sirin E, Parsia B, Grau B C, Kalyanpur A, and Katz Y 2007 Pellet: a Practical OWL-DL Reasoner. *Web Semantics: Science, Services and Agents on the World Wide Web,* 5, 51–53

Tong X, Liang D, and Jin Y 2014 A linear road object matching method for conflation based on optimization and logistic regression. *International Journal of Geographical Information Science,* 28 (4), 824–846

Tong X, Shi W, and Deng S 2009 A probability-based multi-measure feature matching method in map conflation. *International Journal of Remote Sensing,* 30 (20), 5453–5472

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

Yang B, Zhang Y, and Lu F 2014 Geometric-based approach for integrating VGI POIs and road networks. *International Journal of Geographical Information Science,* 28 (1), 126–147

Yang B, Zhang Y, and Luan X 2013 A probabilistic relaxation approach for matching road networks. *International Journal of Geographical Information Science,* 27 (2), 319–338

Table 1: Data used for Nottingham case study

OSM geometry	OSGB geometry	OSM spatial object	OSGB spatial object
953	7795	281	13204

Table 2: Matching OSM spatial objects to OSGB, Nottingham case

σ	TP	FP	TN	FN	recall	precision
1	1	0	72	208	0.005	1
3	25	1	72	183	0.12	0.96
5	100	9	68	104	0.48	0.92
10	162	17	67	35	0.78	0.91
15	173	20	64	24	0.83	0.90
20	175	21	64	21	0.84	0.89
30	177	21	65	18	0.85	0.89
40	177	36	54	14	0.85	0.83
50	178	37	53	13	0.85	0.83
60	178	39	52	12	0.85	0.82
70	178	39	52	12	0.85	0.82
80	178	44	47	12	0.85	0.80

Table 3: Data used for Paris case study

OSM geometry	OSGB geometry	OSM spatial object	OSGB spatial object
4712	4776	326	29

Table 4: Matching OSM spatial objects to IGN, Paris case, σ = 40 meters

TP	FP	TN	FN	precision	recall
15	2	309	2	0.88	0.83