PATTERN CLASSIFICATION APPROACHES TO MATCHING BUILDING POLYGONS AT MULTIPLE SCALES

Xiang Zhang [a, b,] *, Xi Zhao [a, b], Martien Molenaar [b], Jantien Stoter [c], Menno-Jan Kraak [b], Tinghua Ai [a]

[a] School of Resource and Environmental Science, Wuhan University, China - tinghuaai@gmail.com
[b] ITC, University of Twente, the Netherlands - {xzhang, xzhao, molenaar, kraak }@itc.nl
[c] Delft University of Technology, the Netherlands - j.e.stoter@tudelft.nl

Commission II, WG II/2

KEY WORDS: Data Matching, Multi-Scale Modeling, Map Generalization, Pattern Classification, Building Feature

ABSTRACT:

Matching of building polygons with different levels of detail is crucial in the maintenance and quality assessment of multi-representation databases. Two general problems need to be addressed in the matching process: (1) Which criteria are suitable? (2) How to effectively combine different criteria to make decisions? This paper mainly focuses on the second issue and views data matching as a supervised pattern classification. Several classifiers (i.e. decision trees, Naive Bayes and support vector machines) are evaluated for the matching task. Four criteria (i.e. position, size, shape and orientation) are used to extract information for these classifiers. Evidence shows that these classifiers outperformed the weighted average approach.

1. INTRODUCTION

Geospatial data are usually collected for the same geographic areas from different sources and/or at different scales, and for different purposes. To make best use of different data sources, e.g., to carry out advanced spatial analysis based on different abstraction levels (Timpf et al., 1992; Lüscher et al., 2009), matching between datasets is needed. On the other hand, to fulfill the increasing and diverse demand of spatial data at various resolutions and scales, detailed spatial databases are being built or under construction via generalization in many countries, from which smaller scale representations can be derived. However, since fully automated generalization is to date not available, multiple representation databases (MRDBs) became a compromise (Hampe et al., 2003; Sarjakoski, 2007). That is, spatial data of different levels of detail are stored simultaneously and updates are propagated across scales. In this process data matching is key to establishing links between corresponding objects for the maintenance (Kilpeläinen, 2000). Additionally, to automatically assess the quality of generalized objects with respect to initial ones, links between corresponding objects are also required (Stoter et al., 2009).

Matching spatial objects from two heterogeneous datasets is a complex decision process. To decide which pairs of objects match or similar, we need different similarity measures and complex reasoning. Two fundamental problems arise. First, what are the key criteria (or variables) that help determine the matching? Second, how can we make a decision based on the multiple criteria?

Previous work has been dedicated to the development of new similarity measures. In general, those measures can be divided into geometric, semantic and contextual measures. For instance, Beeri et al. (2005) developed spatial join algorithms that match points only using their locations. To match more complex objects (polygons and networks), other geometric information such as angles, shapes, topological properties are also used (Walter and Fritsch, 1999; Gösseln and Sester, 2004). Some other matching approaches also compare the semantics of objects, especially names (Raimond and Mustière, 2008), provided that the attribute was collected for the datasets.

A remarkable approach, proposed by Samal et al. (2004), measures the contextual similarity between two buildings. The context (i.e. surrounding landmarks) of an object is captured in a proximity graphs, and the contextual similarity is calculated between two graphs using displacement vectors. In view of this, Kim et al. (2010) represent context (also landmarks) by a triangulation structure, where the contextual similarity is measured based on areas and perimeters of the triangles organized around the building. This method is more reliable in case of large discrepancies existing between matching datasets; a limitation is that the matching of landmarks relies entirely on names, which is less applicable since names are not always available in topographic data. Note that, to use context one should either match the context, as did in Samal et al. (2004), or refer to a unique context to which both datasets refers.

On the other hand, combining various matching criteria into a decision is still a challenge. Approaches based on a single criterion (e.g. Kim et al., 2010) are free from this issue. However, single source of information does not provide enough evidence for a reliable decision. Therefore, we claim that data matching should combine multiple sources of information as evidence to improve the matching.

This paper aims to tackle this multivariate decision problem. A straightforward approach to this is weighted average. This approach is commonly used (e.g. Walter and Fritsch, 1999; Samal et al., 2004) and consists of two steps: (1) normalizing measured values, and (2) assigning weights to different measures. Clearly, both steps can be problematic. For one thing, normalization factors may not always be available. For another, manual weighting is usually subjective; even experts may sometimes fail to assign appropriate weights. Additionally, as

* Corresponding author. xzhang@itc.nl; xiang.zhang@whu.edu.cn

data matching is essentially an uncertain process, crisp decisions would inevitably reduce the matching performance.

To address these issues, we fit the data matching into a pattern classification framework, which are particularly effective in solving multivariate decision problems. One advantage is that the model parameters can be learned from available data, and the subjective weighting can hence be avoided. Moreover, some of the classification methods (e.g. probabilistic ones) can handle uncertainties, which may help to improve decisions in ambiguous situations.

The remainder of this paper is organized as follows. Section 2.1 formalizes the data matching into a pattern classification problem; then basic geometric criteria and supervised classifiers are briefly described in Sections 2.2 and 2.3. An extension is presented in Section 2.4 which integrates soft classification and domain knowledge to improve the matching. The classifiers are evaluated and discussed in Section 3. This paper ends with conclusions in Section 4.

2. DATA MATCHING AS PATTERN CLASSIFICATION

2.1 Problem Formalization

Data matching aims to find all possible correspondence pairs from two datasets based on several criteria. Each criterion compares a specific characteristic (e.g. shape or orientation) between a pair of objects and yields a measured value. Based on the measured values a decision can be made as to whether this pair of object matches or not. In the following, we formalize this problem as a pattern classification problem.

Let $r_{ij} = (d_i, g_j) \in D \times G$ be a relation or pair of objects, where $d_i \in D$ and $g_j \in G$ are objects in detailed and generalized data. Data matching can then be viewed as a two-category pattern classification problem with category C = {'Matched', 'UnMatched'}. In other words, r_{ij} can be classified into a category $c_k \in C$, depending on the feature vector or measured characteristics $(f\,|\,f_1, \ldots, f_n)$.

Formally, there exists an unknown function $g : D\times G \rightarrow C$ that maps an input pattern $(r_{ij}; f)$ to a category label c_k. However, since such an ideal function is not available for real applications, most classification approaches learn from training patterns $TP = \{(f_1, c_1), \ldots, (f_n, c_n)\}$ and produce a function h that approximate g as closely as possible (supervised learning).

2.2 Basic Criteria

Four criteria, i.e., position, size, shape, and orientation similarity, are used based on commonsense knowledge to show the potential of classification based matching. These criteria are measured from pairs of buildings (i.e. d_i and g_j from detailed and generalized datasets).

First, position similarity is measured based on distance between building centroids. Second, we define size similarity based on the following size ratio:

$$\mathrm{SizeSim}(d_i, g_j) = \mathrm{Area}(d_i) \big/ \mathrm{Area}(g_j) \tag{1}$$

where d_i and g_j represent buildings in different datasets. This size similarity is interpreted as follows: when SizeSim(·) more approaches to 1, the two buildings are more similar in size.

Shape of buildings is characterized by *shape index* (Peter, 2001) which is formally defined:

$$\mathrm{ShapeIndex}(p_i) = \frac{\mathrm{Perimeter}(p_i)}{2 * \sqrt{\pi * \mathrm{Area}(p_i)}} \tag{2}$$

where p_i is a polygon. Shape index measures the complexity (compactness) of shapes with respect to circle. The ratio of *shape index* is used to compare the relative complexity of two shapes:

$$\mathrm{ShapeSim}(d_i, g_j) = \mathrm{ShapeIndex}(d_i) \big/ \mathrm{ShapeIndex}(g_j) \tag{3}$$

The measure of building orientation is based on wall statistical weighting (WSW) described in Duchêne et al. (2003). The resulting orientation is wall direction α in [0, π/2]. The result also comes with a confidence value (numbers indicated in Figure 1a), which is calculated by counting the proportion of length of the edges that orient to α ± σ (a tolerance). Typical buildings have two perpendicular wall directions (α and α + π/2). The walls of direction α + π/2 also add to the confidence value of the resulting WSW orientation α (e.g. confidence values of A - E in Figure 1a approach to 1).

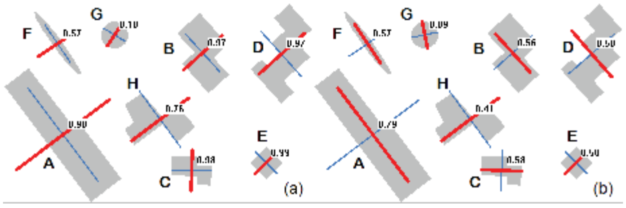


Figure 1: Building orientation measures: (a) wall statistical weighting; (b) adaptation in this approach (output orientations are in bold lines with confidence values numbered upper-right)

In this paper, we adapted the original WSW, in which we distinguish wall direction α from α + π/2. The output orientation is the direction (in [0, π]) in which lengths of walls accumulate most; in most cases this is the dominant one (i.e. major wall direction) of the two perpendicular directions. After adaptation the output orientations adjust better to their major wall directions (e.g. A, B, C in Figure 1b). Confidence value decreases accordingly in our adaption since walls of direction α + π/2 do not add to walls of α (e.g. A - E, H in Figure 1b). The adapted confidence value is now correlated with the degree of elongation (strength of major wall direction). A square (E) with a weak major wall direction has a low confidence (0.5); an oval (F) with a strong major wall direction has a relatively higher confidence (0.57). Note that this adaption is by no means to describe a general orientation, but to choose from among the wall directions the most significant one (in [0, π]). However, except for round shapes (G) the adapted WSW is sufficient for measuring the similarity of building orientations even in the case of stair-like shapes (D). After generalization, stair-like shapes should remain their wall directions to keep their characteristics, but their general orientations may change.

$$\begin{cases} \text{Dev}(d_i, g_j) = \left|\text{WSW}(d_i) - \text{WSW}(g_j)\right|, \\ \text{Dev}(d_i, g_j) = \pi - \text{Dev}(d_i, g_j), \text{if Dev}(d_i, g_j) > \pi/2 \end{cases} \tag{4}$$

In Equation (4), we define similarity by orientation deviation between two buildings. The smaller the deviation is the two buildings are more similar.

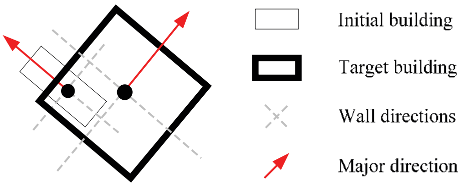


Figure 2: Initial and correspondence buildings with strong and weak major directions

Considering the fact that some buildings have very strong major wall directions (e.g. the initial building in Figure 2) and others may have very weak ones (e.g. the target building in Figure 2), calculation based on Equation (4) may result in an orientation deviation ≥ π/2, indicating that the two are very different in orientation. This is however not true as shown in Figure 2.

To better account for this, we use a confidence threshold T_C to distinguish between strong and weak major wall directions. If the confidence value is less than T_C the building is regarded as having weak major wall direction, and vice versa. Further, if at least one of the matching candidates has a weak major wall direction, the orientation difference between these two should not exceed π/4; only if both buildings in the pair have strong major wall directions, the orientation difference is based on Equation (4). The new function is defined as follows:

$$\text{OriDiff}(d_i, g_j) = \begin{cases} \pi/2 - \text{Dev}(d_i, g_j), & \text{if Dev}(d_i, g_j) > \pi/4 \wedge \\ & (\text{Con}(d_i) \le T_C \vee \\ & \text{Con}(g_j) \le T_C) \\ \text{Dev}(d_i, g_j), & \text{otherwise} \end{cases} \tag{5}$$

Con(·) is the confidence value of object orientation. In this study, $T_C = 0.55$ was empirically determined from training data.

2.3 Supervised Classifiers

In this section, we briefly describe the supervised classifiers we tested in this research. These are classification and regression tree (CART) (Breiman et al., 1984), C4.5 algorithms (Quinlan, 1993), Naive Bayes classifier (a probabilistic model) and Support Vector Machines (SVM). Detailed treatments can be found in Duda et al. (2001).

Different parameters that are used to tune the above-mentioned classifiers are briefly introduced here. First, for decision trees, we used a rule for CART to stop splitting when the majority class rate reaches MR(%). So we use CART^{MR} to denote its different versions, with CART^{*} denoting no stopping rule applied. Then, a Radial Basis Function (RBF) kernel is used for SVM, where the best combination of C > 0 (penalty parameter of classification error) and the kernel parameter γ should be found for an optimal classification. The optimal combination was automatically learnt from training data using LIBSVM[1] package with a 10-fold cross-validation. Hence, we use $\text{SVM}^{\text{C},\gamma}$ to denote different combinations of parameter values.

2.4 Incorporating Domain Knowledge

The generalization knowledge can be used to improve the classification results. Note that the knowledge can only be integrated with classifiers that can handle uncertainties (e.g. Naive Bayes). There are two basic rules:

- Rule 1: any generalized object should link to at least one initial object;
- Rule 2: any initial object should link to one most probable generalized object.

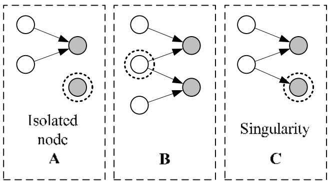


Figure 3: links between initial (white nodes) and generalized (dark nodes) objects

We further distinguish between three ambiguous situations (Figure 3) where the above-mentioned rules are violated:

A. No initial buildings is linked to the generalized building (this building is called *isolated node*);

B. In a *cluster* (group of objects connected by the links), initial buildings have more than one link to generalized buildings;

C. Similar to case B; but differently, only one initial building is linked to the generalized building, creating a *singularity*.

These situations can be improved by the following step-by-step refinement:

1. Link every isolated node (Figure 4a) with the most probable candidate;
2. For each cluster, if there is no singularity (Figure 4b), select the most probable link from initial buildings and remove less probable ones;
3. Otherwise, for each identified singularity s_i, cut all links from initial building d_i except for the link between (d_i, s_i) and update the cluster;
4. Repeat step 2 and 3 until none of the above tree situations can be found.

3. EXPERIMENTS AND DISCUSSION

We implemented the described work as follows. First, the four measures were implemented based on *GenTool* – an interactive generalization and evaluation system developed by a group of colleagues at Wuhan University, China. Training samples were generated in *GenTool* and exported to classifiers. The classifiers were implemented using third party software packages. Specifically, Naive Bayes classifier and CART were realized in MATLAB® software[2]; C4.5 was implemented based on the code provided by Dr. Ross Quinlan[3] (inventor of C4.5);

[1] LIBSVM: http://www.csie.ntu.edu.tw/~cjlin/libsvm/

[2] MATLAB 7.8 (R2009a): http://www.mathworks.com/

[3] http://www.rulequest.com/Personal/

LIBSVM package was used for SVM implementation. Additionally, an interactive matching toolbox and a weighted average approach were implemented to help human operators generate training data and to compare the performance of the classifiers with respect to the weighted average approach.

3.1 Training Data

The datasets to be matched are Dutch topographic datasets at 1:10k and 1:50k (i.e. TOP10NL and TOP50vector, Kadaster). Four datasets were used here, i.e., TOP10NL and TOP50vector at study area A and B, respectively. Study area A represents an area which is characterized by suburbs and rural settlements mixed with a small portion of towns; whereas study area B shows a rather pure characteristic which is dominated by small towns. Another aim is to show whether different characteristics of data influence the matching accuracy. The training samples were generated by experts and are summarized in Table 1.

	No. of building		Training sample
	Scale	Scale	No. of pairs
	1:10k	1:50k	(Matched | UnMatched)
Area A	4272	1720	8934 (1678 | 7256)
Area B	2646	1637	9445 (1774 | 7671)

Table 1: Overview of topographic data and training samples

3.2 Evaluation Procedure and Criteria

The following experiments were carried out. First, the classifiers were trained with one training set (either A or B), and tested with the same set or using a 10-fold cross-validation (results are not shown due to limited space). Second, to show how well the trained classifiers can be applied to classify novel patterns (unknown data), we trained the classifiers with sample A and tested with B, and then reverse. This way we also get insight into whether the prediction power of the classifiers relies on spatial characteristics. To classify unknown data, for each source object at the target scale we select candidates in the initial data that falls into some radius (R) of the source; R was empirically determined to cover potential candidates for a given dataset. Multiple matched candidates are conditionally regarded as n-to-1 matching (see Section 3.4). Different versions of the classifiers were evaluated, including CART*, CART$^{95\%}$, CART$^{85\%}$, SVM$^{0.5,2}$ (see Section 2.3). The criteria used to evaluate the performance of these classifiers are *precision* and *recall*. Besides, tree size is used to evaluate decision trees.

3.3 Classification Accuracy

To summarize, training a classifier and testing it with the same data obtained higher precision and recall than train it with one and predict on another. For example, C4.5 obtained 87.7% precision and 88% recall, which is better than its performance shown in Table 2, to name but a few. CART*, in particular, obtained about 94% precision and 96% recall when trained and tested with the same data. This probably means an over-fitted model.

	Setting 1: classifier trained with A and tested with B		
Classifier	Precision [%]	Recall [%]	Tree size [leaf no.]
CART*	37.5	78.8	242
CART$^{95\%}$	85.9	75.7	128
CART$^{85\%}$	83.9	80.1	45
C4.5	84.9	81.0	19
NB	84.9	82.0	N/A
SVM$^{0.5,2}$	85.3	79.0	N/A
	Setting 2: classifier trained with B and tested with A		
Classifier	Precision [%]	Recall [%]	Tree size [leaf no.]
CART*	75.8	83.3	325
CART$^{95\%}$	82.1	82.4	186
CART$^{85\%}$	80.1	86.3	94
C4.5	85.9	82.9	11
NB	79.0	87.7	N/A
SVM$^{0.5,2}$	81.5	88.2	N/A

Table 2: Performance of different classifiers and for two settings

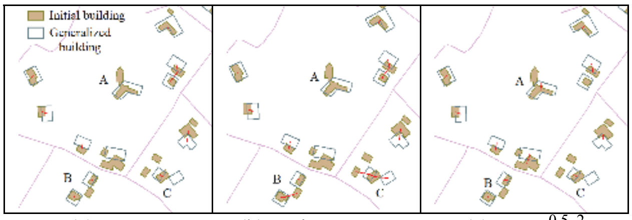
	Precision	Recall
Weighted average	61.7%	61.7%

Table 3: Performance of weighted average approach on study area A with normalized and equally weighted measures

The prediction capability of the trained classifiers on new data is shown in Table 2. Table 2 shows that most classifiers worked satisfactorily for both settings and outperformed the weighted average approach (Table 3), expect for CART*. In general, decision trees provide more interpretable results (i.e. rules) than numerical learning. In addition, Table 2 confirms that higher performance in classifying new data is correlated to relatively smaller sizes of generated trees. Among other decision trees, C4.5 appears to be the most promising in this matching task due to its better performance, its stability in reversing training and test sample and its more tractable tree sizes. CART* performed poorly because it over fitted the training samples (see also our discussion in the previous paragraph) and generated over complicated trees, which not only makes the resulting rules more difficult to interpret but also reduces their performance in classifying novel patterns. Concerning C4.5, NB and SVM ($C = 0.5$, $\gamma = 2$ automatically computed for training samples), no persistent conclusion can be drawn as to the difference in their performance. It is however known that classification accuracy of NB classifier can be further improved (Section 3.4).

Besides, higher precisions and lower recalls can be observed for the classifiers trained with dataset A (characterized by a mixture of towns, suburb and rural settlements) and tested with dataset B (characterized mainly by towns) compared with the reverse setting. Note that both training sets were carefully prepared to gain the same positive class rate (Table 1). This ensures that such a difference was not caused by different positive class rates of the training samples. This suggests that spatial characteristics of the data have an impact on classification performance but not too big. However, how different characteristics may affect the matching accuracy needs to be further investigated.

3.4 Improvement by domain knowledge



(a) C4.5 (b) Naive Bayes (c) SVM$^{0.5,\,2}$

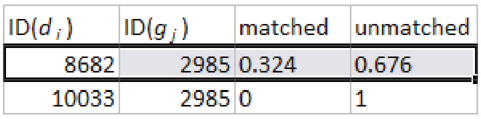
Figure 4: Matching examples predicted for sample set A by training from set B (links are shown in red between initial and generalized buildings)

A closer look at the matching results gets insight into where and why misclassifications occurred. Several typical poor matching was identified:

- Some pairs of buildings (should be matched) were unmatched because their shapes are drastically different given that they are very close (e.g. A in Figure 4a and 4b);
- Some pairs (should not matched) were mismatched because their shape, orientation and position are too similar (e.g. B in Figure 4b);
- Incorrect matching between groups of buildings where n:m relations are likely to happen (e.g. C in Figure 4a-c); in this case, contextual information should improve the matching.

Some of the above misclassifications can be improved by incorporating the generalization knowledge as described in Section 2.4. For example, the poor situations A, B, C in Figure 4b) by Naive Bayes are improved in the following way.

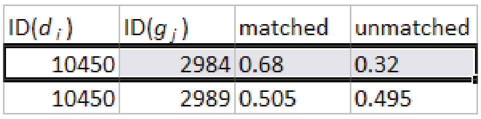
Case A: Two related pairs are searched in the probability table (Figure 5); the generalized object (#2985), with two potential links that were labeled by NB as UnMatched. According to Rule 1, a link with relatively higher positive probability is selected as best fitted link. The selected building (#8682) proves to be the correct correspondence.



ID(d_i)	ID(g_j)	matched	unmatched
8682	2985	0.324	0.676
10033	2985	0	1

Figure 5: Probability table for case A

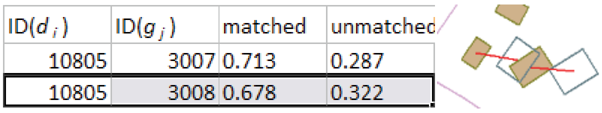
Case B: One initial building (#10450) has two correspondences in generalized dataset, which violates Rule 2. Since no singularity is found in this cluster, Rule 2 can be applied directly by removing one of the links and the most probable link is selected (Figure 6). The selected correspondence (#2984) is the upper one in the cluster B in Figure 4b, which is a more reasonable result.



ID(d_i)	ID(g_j)	matched	unmatched
10450	2984	0.68	0.32
10450	2989	0.505	0.495

Figure 6: Probability table for case B

Case C: It is more complex as a singularity is detected (the right most one in cluster C in Figure 4b). The detected singularity (#3008) links to the building (#10805) in initial dataset (Figure 7a), therefore this link has to be kept. Meanwhile, the other outgoing link from #10805 should be removed according to Rule 2, though it appears to be a more probable link for #10805. After this, the matching result is as follows (Figure 7b), and surprisingly this is exactly what the manual matching was like, even without the use of contextual information.



ID(d_i)	ID(g_j)	matched	unmatched
10805	3007	0.713	0.287
10805	3008	0.678	0.322

(a) (b)

Figure 7: Probability table for case C (a) and the result after domain knowledge is considered (b)

The pair-wise matching allows for n-to-1 and n-to-m relationship to be implicitly modeled (e.g. {{a1,b1}, {a2,b1}, {a2,b2}, {a3,b2}} forms a 3-to-2 relationship). However, current use of domain knowledge (Rule 2) as a post-process is to detect and remove incorrect relationship such as the group C in Figure 4b, which naturally disallows n-to-m relationships (although n-to-1 is still allowed). Better rules are required to replace Rule 2 in order to distinguish incorrect correspondence and potential n-to-m relationships. A prior matching of building groups as described in Zhang et al. (2010) may be helpful.

In summary, classifiers with probability structures and soft decisions are more promising in the matching as domain knowledge can be incorporated to improve the performance. As described above, the matching results obtained from Naive Bayes can be improved by further analyzing the probabilities using the domain knowledge (Section 2.4). Traditional SVM as used here only provides crisp decisions. However, if a probabilistic SVM (Platt, 1999) is used, the domain knowledge can also be incorporated to improve the SVM-based matching.

3.5 Reflection on the matching criteria

This paper presents a first attempt into the classification-based approach to data matching where multivariate decision is important. So the selection of optimal criteria (and measures) to achieve the best possible matching results was not the focus. Four categories of criteria (position, size, shape and orientation) were used based on commonsense knowledge. A correlation analysis (as in Werder et al., 2010) was later carried out which shows no significant correlation between the four measures. However, one should note that the categories are by no means complete and the measures used to evaluate the criteria may not be the optimal ones.

For example, it is questionable whether to use the size ratio for the matching because different size ratios can be caused by enlarging smaller objects, though a distribution of size ratios can be learnt which may facilitate the classification. To get more insights, we carried out parallel experiments where the size criterion was removed. For setting 1 we found that for C4.5, NB and SVM precision decreases and recall increases, indicating that while more true positives (correct links) were found, even more false positives were also produced, which is arguably undesirable. For CART of different versions both precision and recall decrease. Similar results were obtained for setting 2. This suggests that the size criterion adds more or less to the matching. However, a redesign of size criterion in the future taking into account the possible change ratio in relation to initial sizes may give more discriminating power.

Likewise, by removing shape respective orientation criteria, obvious decrease in both precision and recall occurs for the classifiers. This suggests that the used measures are relevant for building matching though better performance can be anticipated by designing measures that differentiate special cases (e.g. oval shapes).

By iteratively removing and adding matching criteria and measures we get an impression of their relative contributions to the matching. Our experiment shows that distance criterion was the dominant parameter for all classifiers, while the contribution of e.g. size and shape varied for different classifiers. However, it is unknown yet whether it is justified to use this approach to study the relative weighting of model parameters. Also as we argue previously, explicit weighing by designer should not be a problem in the supervised classification approach.

In future research, more measures should be analyzed for different criterion categories, the optimal one or combination of ones can be chosen using techniques such as principal component analysis (Burghardt and Steiniger, 2005). Further, other criterion categories such as semantic and contextual ones can be integrated to improve the data matching.

4. CONCLUSION

Fitting data matching process into a pattern classification framework aims to provide a more generic approach to the matching of spatial objects (polygons, linear features, networks, etc.). In this framework, combining multiple criteria into final decisions is more effective and adaptive: rather than arbitrary normalization and weighting, model parameters can be learned from training data.

Four classifiers (CART, C4.5, Naive Bayes and SVM) with different parameter values were tested to show their possibilities in matching building polygons. They outperformed weighted average in terms of classification accuracy. Generally, the accuracy (both precision and recall) reached approximately 80% and higher, based on four simple similarity measures (i.e. position, size, shape and orientation). To further improve the matching result, advanced measures like semantic and contextual similarity should be considered. Moreover, classifiers that can handle uncertainties could be further improved by integrating domain knowledge.

REFERENCES

Beeri, C., Doytsher,Y., Kanza,Y., Safra,E. and Sagiv,Y., 2005. Finding corresponding objects when integrating several geo-spatial datasets. In: *Proceedings of the 13th ACMGIS*, pp. 87-96.

Breiman, L., Friedman, J. H., Olshen, R. A., and Stone, C. J., 1984. *Classification and regression trees*. Monterey, CA: Wadsworth & Brooks/Cole Advanced Books & Software.

Burghardt, D., and Steiniger, S. 2005. Usage of Principal Component Analysis in the Process of Automated Generalisation. In: Proceedings of 22nd ICC (A Coruña, Spain).

Duchêne, C., Bard, S., Barillot, X., Ruas, A., Trévisan, J. & Holzapfel, F., 2003. Quantitative and qualitative description of building orientation. In *the 5th ICA Workshop on Progress in Automated Map Generalization*, 10p.

Duda, R.O., Hart, P.E. and Stork, D.G., 2001. *Pattern Classification (2nd edn)*. Wiley-Interscience Publication, New York, 654p.

Gösseln, G.v. and Sester, M., 2004. Integration of geoscientific datasets and the German digital map using a matching approach. In: *Proceedings of the XXth International Society for Photogrammetry and Remote Sensing Congress*, pp. 1249-1254.

Hampe, M., Anders, K. and Sester, M., 2003. MRDB applications for data revision and real-time generalization. In: *Proceedings of the 21st ICC*, pp 192-202.

Kim, J., Yu, K., Heo, J., and Lee, W., 2010. A new method for matching objects in two different geospatial datasets based on the geographic context. *Computers & Geosciences*, 36, pp. 1115-1122.

Kilpeläinen, T., 2000. Maintenance of Multiple Representation Databases for Topographic Data. *Cartographic Journal, The*, 37(2), pp. 101-107 .

Lüscher, P., Weibel, R., and Burghardt, D., 2009. Integrating ontological modelling and Bayesian inference for pattern classification in topographic vector data. *Computers, Enviroment and Urban Systems*, 33, pp. 363-374.

Mustière, S. and Devogele, T., 2008. Matching Networks with Different Levels of Detail. *Geoinformatica*, 12, pp. 435-453.

Peter, B. and Weibel, R., 2001. Using vector and raster-based techniques in categorical map generalization. In *the 3rd ICA Workshop on Progress in Automated Map Generalization*, 14p.

Platt, J., 1999. Probabilistic outputs for support vector machines and comparison to regularized likelihood methods. In *Advances in Large Margin Classifiers*, MIT Press, pp. 61-74.

Quinlan, J.R., 1993. *C4.5: Programs for machine learning*. Morgan Kaufmann, San Francisco, CA.

Raimond, A.and Mustière, S., 2008. Data Matching - a Matter of Belief. In: *Headway in Spatial Data Handling*, pp. 501-519.

Samal, A., Seth, S. and Cueto, K., 2004. A feature-based approach to conflation of geospatial source. *International Journal of Geographical Information Science*, 18(5), pp. 459-489.

Sarjakoski, L. T., 2007. Conceptual models of generalisation and multiple representation. In: *Generalisation of Geographic Information: Cartographic Modelling and Applications*, Series of International Cartographic Association, Elsevier, pp. 11-35.

Stoter, J., Burghardt, D., Duchêne, C., Baella, B., Bakker, N., Blok, C., Pla, M., Regnauld, N., Touya, G. and Schmid, S., 2009. Methodology for evaluating automated map generalization in commercial software. *Computers, Environment and Urban Systems*, 33(5), pp. 311-324.

Timpf, S., Volta, G., Pollock, D. and Egenhofer, M.J., 1992. A conceptual model of wayfinding using multiple levels of abstraction. In: *Theories and Methods of Spatio-Temporal Reasoning in Geographic Space*, Springer, pp. 348-367.

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

Werder, S., Kieler, B., and Sester, M., 2010. Semi-automatic interpretation of buildings and settlement areas in user-generated spatial data. In: *the 18th ACMGIS*, pp. 330-339.

Zhang, X., Stoter, J., Ai, T., and Kraak, M.-J., 2010. Formalization and data enrichment for automated evaluation of building pattern preservation. In: *Joint International Conference on Theory, Data Handling and Modelling in GeoSpatial Information Science (SDH2010)*, volume XXXVIII, Part 2, pp. 267–272.