

Towards Registration of Construction Drawings to Building Information Models using Knowledge-based Extended Geometric Hashing

Maciej Trzeciak, André Borrmann
Technical University of Munich, Germany
maciej.trzeciak@tum.de

Despite the increasing adoption of BIM technologies, construction drawings are still excessively used in the industry. Many leading AEC figures predict the coexistence of 3D models and 2D drawings for at least a decade. Very often, these two are managed – and in particular for the more advanced design stages – developed separately from each other which puts their consistency at severe risk. In current practice, the deviation detection between drawings and 3D models is a manual, cumbersome, and erroneous process. To overcome this issue, the need for automation emerges. In this paper, we explore this problem and propose a robust drawing-to-model registration system which provides a basis for subsequent consistency verification. It translates the 2D-3D matching problem into the 2D-2D one by cutting a 3D model into sections and measuring similarity between them and the construction drawing. At the core of the proposed approach is geometric hashing, which we extend to further automate the registration process. We propose to build two geometric hash tables which are being matched against each other while measuring similarity. In order to reduce the matching ambiguity – which is the drawback of the proposed extension – we estimate the minimal number of bases to be randomly sampled in the presence of outliers by using the random sample consensus. Moreover, we decrease the burden of building hash tables by reducing the bases – in which features are described – to line segments. We demonstrate the applicability of the proposed system in a case study and provide experimental results.

1. Introduction

The construction industry is currently in a transition period in which drawing-based practices are slowly being replaced by model-based ones, which implies the coexistence of construction drawings and digital 3D models. According to many leading figures in the AEC industry, the intense use of technical drawings in the design and construction of buildings will continue for at least a decade despite the increasing adoption of BIM technologies. In particular, drawings will prevail for a number of years as the legally binding documents for the actual construction processes on site, but also for approval processes by permission authorities. This is also reflected by the recently published international standard ISO 19650 that defines the concept of information containers composed of models and the associated drawings (ISO, 2018). Most of the exchange information requirements (EIR) documents and the associated contracts demand the consistence of the models and the drawings handed over at a certain point in time (data drop) (ISO/DIS, n.d.). In many cases, checking the consistency is part of the approval processes for passing one of the quality gates (Preidel *et al.*, 2017). However currently, these checks must be done manually as no automated methods exist.

Commercial BIM design applications allow to derive drawings that are fully consistent with the created BIM model. However, due to the established processes of the industry, construction drawings and BIM models are subsequently managed separately from each other. Even worse, in many cases they are developed further and refined independently from one another. Therefore, the need for automated verification of consistency between BIM models and the corresponding construction drawings persists. This paper does not question the well-documented value of model-based design and construction (Sacks *et al.*, 2019). However, it respects the reality of the coexistence of models and drawings for many years to come. More

specifically, it connects to today's established working practices where drawings form the major basis of information exchange, even if BIM models are created by individual stakeholders. In this paper, we introduce a robust method for the registration of construction drawings to BIM models, which provides the basis for consistency verification and deviation detection.

In a broader sense, shape registration is a task of aligning two shapes in a shared coordinate system (Shao *et al.*, 2011). More specifically, the positioning of a 3D model against an image is known as the model-to-image registration (Jung *et al.*, 2016), co-registration (Avbelj *et al.*, 2015), or 3D-to-2D registration (Wunsch and Hirzinger, 1996). Generally, most registration problems can be divided into three steps: (1) Feature extraction – where distinctive features are derived from two objects to be matched; (2) Matching – which measures the degree of similarity between the two sets of extracted features and (if necessary) establishes the explicit correspondences between them; (3) Transformation – which transforms the matched features so that they align in a shared coordinate system; The researches related to the matching step focus mainly on either feature-based shape matching or matching topology.

We propose a registration system, tailored specifically to the drawing-to-model matching problem in the construction industry, which focuses on feature-based shape matching. The idea of this approach is similar to the one used in view-based 3D model retrieval domain (Tangelder and Veltkamp, 2008; Liu, 2012), in which two 3D models are defined to be similar if they look similar from different views based on the shape of the object in these views. Accordingly, the 2D-3D matching problem is translated into the 2D-2D one.

The ultimate objective of our method is that a given drawing (subsequently called a query drawing) is supposed to be placed (translated, rotated, scaled) in the three dimensional space of a BIM model such that the cross-section derived at the corresponding plane (subsequently called a model section) is the most similar to the query drawing. Before this, however, we derive model sections and measure their level of similarity to the query drawing by comparing the extracted features. For the most similar model section, we establish explicit correspondences to the query drawing and align the drawing to the model section.

At the core of the proposed matching algorithm is geometric hashing. We decrease the burden of building hash tables by reducing the space of bases to line segments. Furthermore, we propose to extend the original geometric hashing algorithm in such a way that the matching process is fully automated. To this end, we propose to build two geometric hash tables which are being matched against each other while measuring similarity between a model section and the query drawing. In order to reduce the matching ambiguity – which is the drawback of the proposed extension – we propose to estimate the minimal number of bases to be randomly sampled in the presence of outliers by using the random sample consensus - RANSAC (Fischler and Bolles, 1981).

The paper has the following structure. Section 2 describes our model-to-drawing registration system, including feature extraction, similarity measure and matching methods, correspondence establishment, and alignment. Section 3 contains a case study for a standard 3-story residential building along with results. We conclude this paper with Section 4 devoted to discussion.

2. Drawing-to-model registration system – methodology

The drawing-to-model registration system focuses on feature-based shape matching and uses knowledge-based extended geometric hashing as the method for measuring similarity and correspondence establishment between a query drawing and a set of cross-sections (in this

paper, we name them model-sections) derived from a 3D BIM model. The overview of these steps is shown in Figure 1. We will walk through them in detail in the next sections.

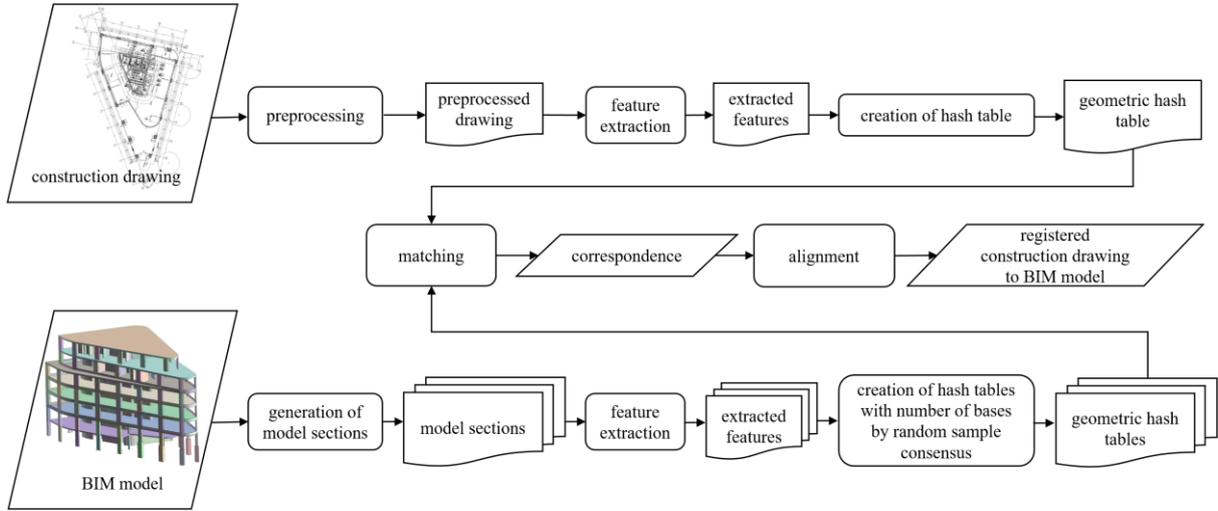


Figure 1: High-level pipeline for the drawing-to-model registration system.

2.1 Generation of model sections

We translate the 2D-to-3D matching problem into 2D-to-2D by cutting the 3D model. In this paper, we use horizontal planes in such a way, that each of them is located above each building level so that it reflects the engineering practice of deriving floor plans at 1.20 m. above a building level. Accordingly, we reduce the number of potential model sections (the discretization of solution space) to be compared with the query drawing significantly. In case of the drawings showing the vertical cross-section of a building, we propose to cut the 3D model vertically and at uniformly distributed distances.

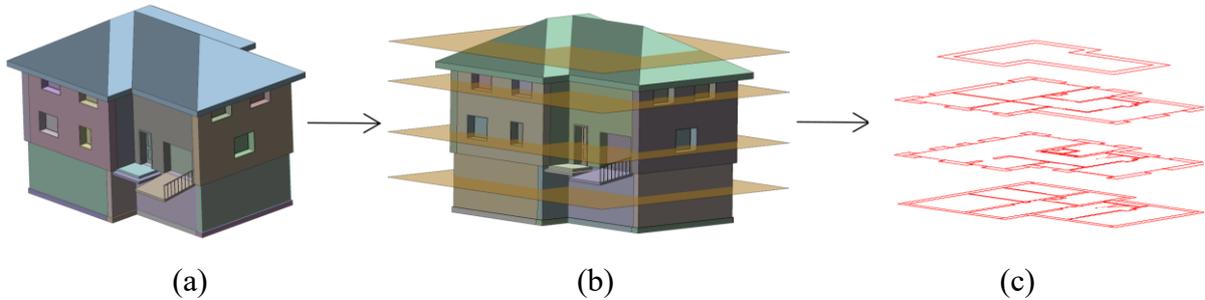


Figure 2: In our system, the 2D-to-3D matching problem is translated into 2D-to-2D by cutting a BIM model (a) using planes as shown in (b). As a result, the solution space is reduced to separate 2D model sections (c).

In case of the drawings of facades, we propose to create orthographic projections of all the sides of the BIM model. Moreover, since BIM models are comprised both of geometry and semantics, we are able to select the types of building elements that we use for matching. For example, elements as doors and windows are shown in construction drawings as icons conforming to engineering standards and not as their pure orthographic projections, and as such, they are not good candidates for the shape-based assessment of similarity by our system.

As shown in Figure 3, the shapes of a floor plan and a model section derived from the same level appear rather similar to a human, however they inevitably differ from each other in details. The purpose of the left one is to convey engineering design and therefore, they are richer in explicit and additional descriptions (hatch patterns, fills, dimensional lines, descriptions) than

BIM models, mainly due to conforming to engineering standards. For example, the building elements such as thermal insulation, concrete columns or walls cut by a plane are filled in with hatch patterns, explicit dimensional lines are drawn, the symbols of certain types of building elements such as doors or windows are agreed icons and differ from their orthographic projections. In addition, what might not be seen instantly from Figure 3 is the fact that the orientation and the scale of the presented drawing and model section are already correct. In a real scenario, the descriptor responsible for measuring similarity must be invariant to rotation, translation and scaling if this process is supposed to be automated and the correspondence properly established. Moreover, construction drawings very often contain additional details not present in BIM models due to the fact that the modeling of all details in 3D is not yet sufficiently efficient.

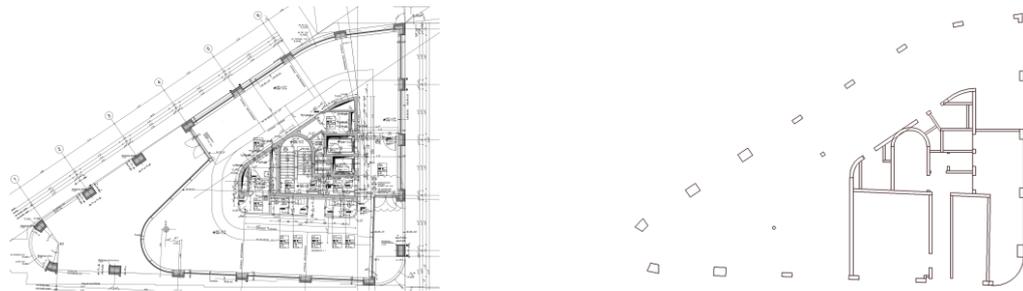


Figure 3: Visual comparison of a construction drawing and its pure section derived from an IFC model at the same level.

Having this in mind, the proposed concept is to measure the commonality of construction drawings and the model sections. The more they have in common, the more similar they are. We extract a unique fingerprint (a unique set of features) occurring in each model section and we find this unique set in the query drawing. In other words, we find the common part of the two sets of features – one set belonging to a model section and the other belonging to the query drawing. We repeat this process for each model section, and the section with the greatest common part is then proposed as the best match. It is then used for the subsequent explicit establishment of correspondence and the alignment.

2.2 Feature extraction

Since the model sections and the floor plan are in vector-based representations, we query them in such a way that we extract their features and segment lines. We do not use methods for detection of key-points or lines from raster images (e.g. Hough line transform).

In building engineering, corner points are the dominant type of features since construction drawings are mainly comprised of straight segments. Accordingly, for each derived 2D model section we extract key-point features as corner points, here understood as the start, end, and intermediate points defining polylines, and as the start and end points of single segment lines whose overlapping points (duplications) are later filtered out. Due to the fact that we propose to build hash tables using segment-wise bases, and not all the possible combinations of bases of key-point features, we extract a set of line segments described by their start and end points.

In a similar vein, we extract the same key-point features and segment lines from the query drawing. In the core of the proposed methodology is the measure of the common part between the model sections and the query drawing. Since the model sections, by definition, contain only elements cut by a plane, and not projected onto the plane, it makes sense to look for these elements in the query drawing. In the engineering convention, such lines cut by a plane are the

thickest, and therefore, we first query the thickest lines in the query drawing, and based on them, we extract the key-point features and line segments. The line segments are ordered sets of points and for each line segment, two bases are later taken into consideration (start-end-point basis and end-start-point basis), which is necessary for unambiguous alignment in a further step of the proposed solution.

2.3 Similarity measurement using geometric hashing

In order to find the common part of two sets of features, the descriptor must be able to partially match objects. By partial matching we mean finding a shape whose part is similar to a part of another shape, which in this case, translates to finding the matching features between two sets. Moreover, it must be tailored for specific purpose of matching many model sections to a query drawing within reasonable time. Accordingly, we do not need descriptors setting up the explicit correspondence for each model section and the query drawing, for example such as presented in (Belongie *et al.*, 2002). Additionally, translation, scaling, and rotation invariances are required simply due to the fact that models and drawings are described in arbitrary coordinate systems. Therefore, in the proposed approach, there is a need to provide the descriptor which supports partial matching and provides fast scores for the common part.

We have decided to match shapes between the corresponding building elements in 2D and 3D, and not – for example – to match the grid systems of drawings and 3D models. From our experience, grid systems are very rarely exported to IFC models, and therefore, matching based on the grids would impose a severe prerequisite. Instead, we provide a general solution to the existing problem which works without major prerequisites and thus, without barriers in its application.

As such, we propose and extend *geometric hashing* which was originally developed as an object recognition technique in computer vision (Wolfson and Rigoutsos, 1997). The general procedure and ideas behind it can be found in the original paper and we skip this part due to the page limit. Geometric hashing consists of two stages – preprocessing and recognition.

Preprocessing phase

In our case, the preprocessing stage turns the features described in segment-wise bases extracted from a query drawing into a hash table. The naïve solution would be to describe the features (n stands for the number of the features) in all their combinations ($C_{n,k}$ – combinations without repetition, with $k = 2$, because a basis consists of an ordered pair of points), which yields the quadratic function as shown in equation 1. The equation shows asymptotic number of naïve bases using all combinations of key-point features needed to build a geometric hash table:

$$C_{n,k} = \binom{n}{k} = \frac{n!}{k!(n-k)!} = \frac{n(n-1)}{2} = O(n^2) \quad (1)$$

We reduce the basis space, however, to segment-wise bases (B_m), meaning that each straight segment becomes a basis. This reduces the number of bases in which a hash table is built to a linear function (see equation 2), as each segment line increases the number of bases by 1 (or in case of query drawings, each segment line increases the number of bases by 2):

$$B_m = 1m = O(m) \quad (2)$$

Recognition phase

Since the hash table of the query drawing is built with the features described in all segment-wise bases, in order to compare the similarity to a model section, we need to sample one

corresponding basis in the model section, and count the number of matching features by accessing entries of the hash table built on the query drawing. In other words, the task comes down mainly to finding a corresponding segment line between the model section and the query drawing. The model section whose basis has the most corresponding features (and thus the greatest common part) is therefore the most similar to the query drawing. In the ideal situation, this task would be trivial if the query drawing and the model section did not have any outliers or noise. We could randomly sample a basis, describe all the features in this basis, and the model section with the highest number of corresponding features would be the most similar to the query drawing. Our model sections, however, do contain outliers because, for example, they are composed of the cross-sections of individual building elements (see an example in Figure 4 which shows a part of a model section comprised of two walls marked in green and blue). Consequently, not all of the line segments between the query drawing and the model section overlap, and those which do not, are considered as outliers (an outlier can be seen in Figure 4 marked with a yellow arrow, whereas the green arrow shows an inlier). Another example where outliers occur is a drawing which consists of building elements not modelled in the BIM model, and therefore not appearing in the derived model sections. Accordingly, there is a need to estimate how many bases need to be sampled from the model section in order to select at least one error-free basis corresponding to the query drawing.

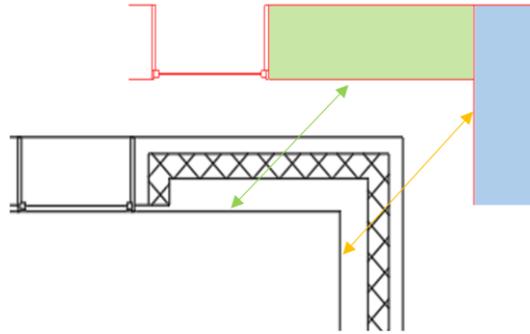


Figure 4: Corresponding segment lines between a model section (at the top) and a query drawing (at the bottom). The presented part of the model section is mainly comprised of two cross-sections of two building elements marked with the green and blue fills. Using a green arrow, we mark an example of bases which correspond with each other (inliers), whereas in yellow, we mark an example of bases which do not correspond with each other (outliers).

The naïve number of the sampled bases would equal to all of the segment-wise bases (as in the preprocessing stage executed on the query drawing). This, however, would increase the burden of building another full hash table, and what is worse, it would increase the matching ambiguity due to redundant bases. In order to reduce these drawbacks, we propose to estimate the minimal number of bases to be randomly sampled in the presence of outliers by using the random sample consensus so that at least one inlier is sampled. Having estimated the minimal number of sampled bases, we describe all the features in these bases and store them in another hash table which is being matched against the hash table of the query drawing. By doing this, we provide the full automation of the recognition phase (and thus the automation of the drawing-to-model registration). The major difference to the original geometric hashing is that the native procedure urges the user to choose an arbitrary basis in the to-be-recognized image, whereas we propose to build another hash table by storing the features described in a certain number of sampled bases (in which there is at least one inlier) estimated by RANSAC, and to match these two hash tables against each other. As a result, the matching bases between the model section and the query drawing can be determined.

Accordingly, we model the outliers as a random variable, and having assumed a certain outlier rate (ϵ), we sample k bases with very high probability (p) to select at least one (r) inlier basis. The number of trials (k) is computed as shown in equation 3 (Fischler and Bolles, 1981).

$$k = \frac{\log(1 - p)}{\log(1 - (1 - \epsilon)^r)} \quad (3)$$

2.4 Correspondence establishment

Geometric hashing does not solve for explicit correspondence between all corresponding features (Belongie *et al.*, 2002). Instead it uses the features to vote for a specific basis. As a consequence, the explicit correspondence in our proposed system can be automatically established only between most voted bases from each hash table.

In our modified version, we simultaneously vote for bases in two hash tables. At the end, there is one basis in each hash table (in case of no redundant bases have been sampled in the recognition phase) that is most voted and we establish the correspondence between them. We do not explicitly recover all the feature-to-feature correspondences. Instead, we use the two corresponding bases between a drawing and a model section in 2D to align them.

2.5 Determination of alignment

Having found two corresponding bases, we align them in the following way: (1) The corresponding segment lines are viewed as vectors and the difference in scale between a model-section and the drawing is determined by the quotient of the norms of these vectors; (2) The translation vector is calculated between the middle points of the corresponding segment lines; (3) The rotation angle is calculated using the formula for the vector-vector dot product, where the corresponding segment lines are seen as vectors. Having calculated the above parameters, the drawing is accordingly scaled, translated and rotated.

3. Test case

We demonstrate the applicability of the proposed system in a case study of a standard 3-story residential building. Our prototype operates on IFC-based BIM models, and construction drawings in the DXF format. The geometric hashing algorithm and its extension has been programmatically implemented from scratch based on the original paper and extended as described in the previous section. IFC and DXF readers, viewers, and scientific computing libraries are free-license packages. The visualization of the BIM model and the floor plan can be seen in Figure 5. The BIM model was created using a commercial BIM modeling tool. The floor plan was then derived and modified to conform to engineering standards using this tool. Ultimately, the model and the floor plan were exported to the IFC and the DXF data formats.

We cut the IFC-based BIM model horizontally at 4 different levels (shown in Figure 6a) in such a way that the horizontal cutting plane is located at 1.20m above each floor level, which translates to the following building levels: -1.30m, 1.20m, 4.00m and 6.20m. The corresponding model sections are shown in Figure 6 b-e. The model sections are comprised of all the instances of parent type `IfcProduct`. The dimensions of the BIM model (and thus model sections) are stored in meters, whereas the dimension of the floor plan are stored in millimetres.



Figure 5: Visualization of a BIM model as a standard 3-story residential building, and its corresponding floor plan portraying the ground floor derived at 1.20m level.

In case of the floor plan, we first filter all the geometric primitives, select those with the highest stroke width and build a geometric hash table for all the extracted key-point features described in all the extracted bases (segment lines). Next, for each model section, we extract its key-point features and segment lines and build a geometric hash table. We calculate the number of randomly sampled bases with the probability for selection at least one error-free base ($p = 0.99$) and outlier rate ($\epsilon = 0.9$). In such a case, the number of sampled bases is equal to $k = 44$, which reduces the number of bases used to build the hash tables around 4-7 times in case of model sections b, c and d presented in Figure 6. In all the cases, we quantize the coordinates of features stored in the hash tables by a bin of size 0.24.

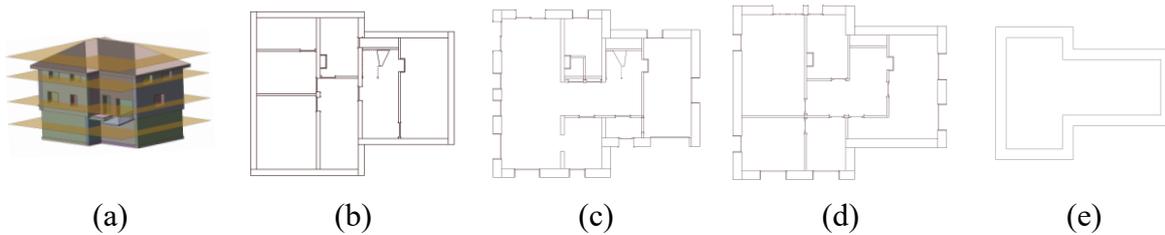


Figure 6: An IFC-based BIM model horizontally cut at 4 different levels – figure a. Figures from b to e present the corresponding model sections.

3.1 Experimental results

We match the hash tables of each model section against the hash table of the query drawing. In Figure 7, we show the 3 most voted bases for each model section. The model section derived at level +1.20 is most voted, and within this level, there are two bases with a very high score. This level is also marked in green on the right side of the figure.

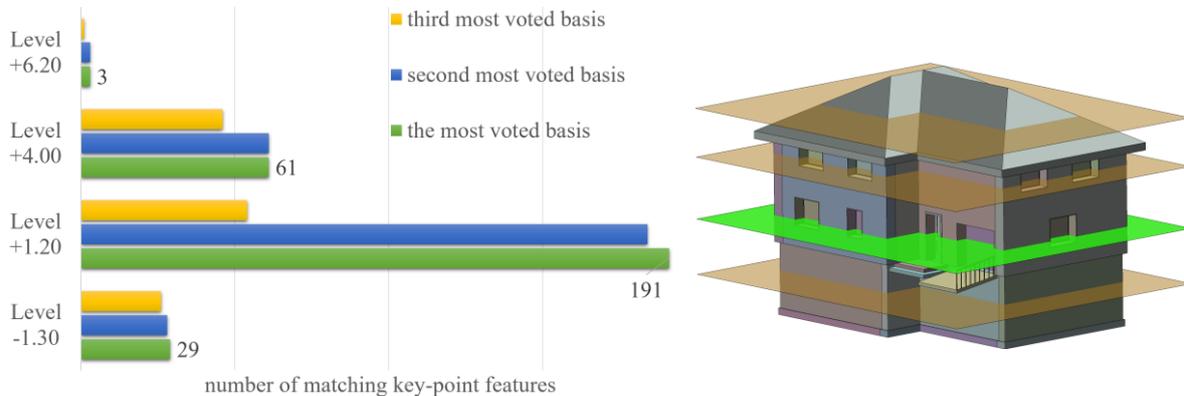


Figure 7: Results obtained after matching the geometric hash table of the query drawing against all the geometric hash tables of the model sections. The first three most voted bases are presented.

The most voted basis at level +1.20 and the most voted basis from the query drawing are then proposed for setting up the correspondence, which is shown in Figure 8 on the left hand side.

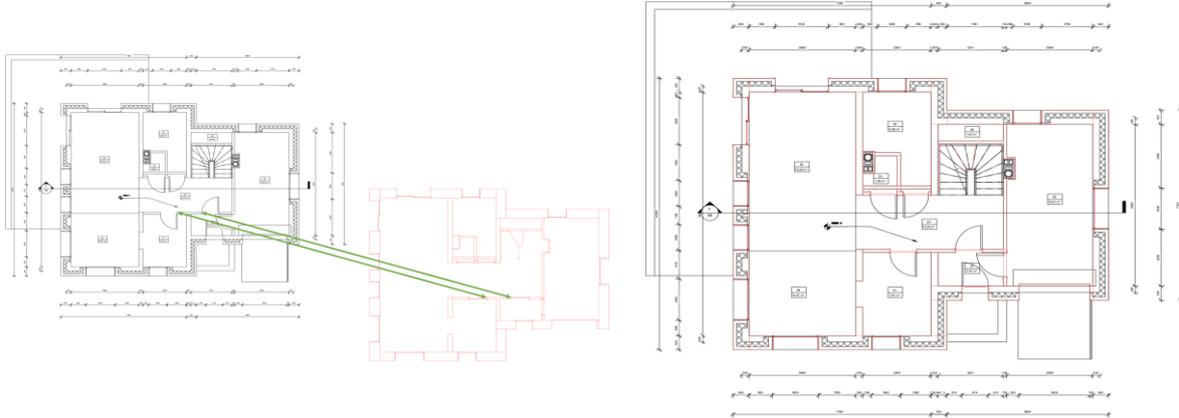


Figure 8: Left: The two corresponding bases between the query drawing and the most voted model section. Right: Aligned query drawing and model section.

4. Conclusions and future work

In this paper, we propose a robust feature-based shape matching method for the registration of construction drawings to BIM models with the extended geometric hashing at its core. We test this method in a case study of a standard 3-story residential building. The early experimental results are promising when it comes to measuring similarity with level-wise accuracy between floor plans in vector-based formats and BIM models using key-point features and segment line bases. The proposed reduction of basis space to segment lines reduces the burden for building hash tables due to its linear (and not quadratic) complexity.

The proposed extension of geometric hashing allows to cast votes simultaneously in two hash tables and therefore, it is possible to automatically establish the correspondence between two most voted bases. The results show that this method correctly attributes the query drawing to one of the model sections, and correctly indicates two corresponding bases which allows to establish the explicit correspondence. As +1.20 level shows, however, the proposed random-sample-consensus-based sampling of bases for model sections in the presence of outliers returned 2 inliers (1 redundant basis). Any redundant basis causes the matching ambiguity while setting up the explicit correspondence between two bases as the margin between the first and the second most voted bases is very little. In the future, in such situations, we propose to measure the similarity between the two registered objects by establishing more corresponding features and calculating the distance between them (for example using the L2 distance function). Moreover, it must be noted that the more similar the model sections are to the query drawing (fewer outliers), the more corresponding key-point features can be matched. In our case, the prototype operates on IFC models, whose pure cross-sections are comprised simply by cutting individual building elements which make the model section be prone to outliers (in comparison to the corresponding construction drawing). Additionally, such parameters as the outlier rate and the bin size need to be tuned on more real projects, as they drive the performance of this system, which we could not show in this paper due to its page limit.

There are many other cases in which our registration system could be tested as well. Matching drawings of details which contain elements not necessarily modeled in BIM models (for example - highly detailed steel connections), matching vertical cross-sections or projections of facades are use cases which could be tested. In all of these cases, the derivation mechanism for model sections presented in this paper must be extended so that it returns good 2D candidates

for comparison to the query drawing. The derivation mechanism is potentially the biggest limitation in the proposed system because it is hard-coded for specific cases (horizontal model-sections for floor plans, vertical models-sections for cross-sections and so on). Also, in case of drawings which are not comprised of mainly straight segment lines, we would have to extend the types of features that are supposed to be matched because the fewer key-point features extracted, the less confident the system is. Therefore, the second major limitation of the proposed system is its non-consideration of curved geometric primitives which limits the applicability of this system in certain cases with more advanced geometry.

Overall, the proposed registration system shows promising results in building engineering for further automation of both the registration process and its subsequent consistency verification which could be used, for example, in the verification of consistency between construction drawings and BIM models at quality gates in approval process by authority.

Acknowledgements

This work is supported by ALLPLAN GmbH and Nemetschek Group. The authors would like to gratefully thank these companies for making this research possible.

References

- Avbelj, J., Iwaszczuk, D., Müller, R., Reinartz, P. and Stilla, U. (2015), “Coregistration refinement of hyperspectral images and DSM: An object-based approach using spectral information”, *ISPRS Journal of Photogrammetry and Remote Sensing*, Vol. 100, pp. 23–34.
- Belongie, S., Malik, J. and Puzicha, J. (2002), “Shape matching and object recognition using shape contexts”, *Pattern Analysis and Machine Intelligence, IEEE*, Vol. 24 No. 24.
- Fischler, M.A. and Bolles, R.C. (1981), “Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography”, *Communications of the ACM*, Vol. 24 No. 6, pp. 381–395.
- ISO (2018), *Organization and digitization of information about buildings and civil engineering works, including building information modelling (BIM): Information management using building information modelling -- Part 1: Concepts and principles* No. ISO 19650-1:2018, available at: <https://www.iso.org/standard/68078.html> (accessed 2019-03).
- ISO/DIS (n.d.), *Information container for data drop: Exchange specification -- Part 1: Container* No. ISO/DIS 21597-1, available at: <https://www.iso.org/standard/74389.html> (accessed 2019-05).
- Jung, J., Sohn, G., Bang, K., Wichmann, A., Armenakis, C. and Kada, M. (2016), “Matching Aerial Images to 3D Building Models Using Context-Based Geometric Hashing”, *Sensors (Basel, Switzerland)*, Vol. 16 No. 6.
- Liu, Q. (2012), *A Survey of Recent View-based 3D Model Retrieval Methods*, available at: <http://arxiv.org/pdf/1208.3670v1>.
- Preidel, C., Daum, S. and Borrmann, A. (2017), “Data retrieval from building information models based on visual programming”, *Visualization in Engineering*, Vol. 5 No. 1, pp. 1–14.
- Sacks, R., Eastman, C.M., Lee, G. and Teichlz, P. (2019), *BIM Handbook: A Guide to Building Information Modeling For Owners, Designers, Engineers, Contractors, and Facility Managers*, Third Edition, Wiley.
- Shao, T., Xu, W., Yin, K., Wang, J., Zhou, K. and Guo, B. (2011), “Discriminative Sketch-based 3D Model Retrieval via Robust Shape Matching”, *Computer Graphics Forum*, Vol. 30 No. 7, pp. 2011–2020.
- Tangelder, J.W.H. and Velkamp, R.C. (2008), “A survey of content based 3D shape retrieval methods”, *Multimedia Tools and Applications*, Vol. 39 No. 3, pp. 441–471.
- Wolfson, H.J. and Rigoutsos, I. (1997), “Geometric hashing: an overview”, *IEEE Computational Science and Engineering*, Vol. 4 No. 4, pp. 10–21.
- Wunsch, P. and Hirzinger, G. (Eds.) (1996), *Registration of CAD-Models to Images by Iterative Inverse Perspective Matching*.