Combining inverse photogrammetry and BIM for automated labeling of construction site images for machine learning

Alex Braun[a,b], André Borrmann[a,b]

[a] *Chair of Computational Modeling and Simulation, Technical University of Munich, Germany*
[b] *Leonhard Obermeyer Center, TUM Center of Digital Methods for the Built Environment*

Abstract

Image-based object detection provides a valuable basis for site information retrieval and construction progress monitoring. Machine learning approaches, such as neural networks, are able to provide reliable detection rates. However, labeling of training data is a tedious and time-consuming process, as it must be performed manually for a substantial number of images. The paper presents a novel method for automatically labeling construction images based on the combination of 4D Building Information Models and an inverse photogrammetry approach. For the reconstruction of point clouds, which are often used for progress monitoring, a large number of pictures are taken from the site. By aligning the Building Information Model and the resulting point cloud, it is possible to project any building element of the BIM model into the acquired pictures. This allows for automated labeling as the semantic information of the element type is provided by the BIM model and can be associated with the respective regions. The labeled data can subsequently be used to train an image-based neural network. Since the exact regions for all elements are defined, labels can be generated for basic tasks like classification as well as more complex tasks like semantic segmentation. To prove the feasibility of the developed methods, the labeling procedure is applied to several real-world construction sites, providing over 30,000 automatically labeled elements. The correctness of the assigned labels has been validated by pixel based area comparison against manual labels.

Keywords: Machine Learning, Labeling, Construction progress monitoring, BIM, Photogrammetry, semantic and temporal knowledge

1. Introduction

Large construction projects require a variety of different manufacturing companies of several trades on site (for example masonry, concrete and metal works, HVAC, ...). An important goal for the main contractor is to keep track of accomplished tasks by subcontractors to maintain the general schedule. Additionally, the documentation of correctly executed tasks plays a crucial role for all involved parties. In construction, process supervision and monitoring is still a mostly analog and manual task. To prove that the work has been completed as defined per contract, all performed tasks have to be monitored and documented. The demand for a complete and detailed monitoring technique rises for large construction sites where the complete construction area becomes too large to monitor by hand, and the number of subcontractors rises. Main contractors that control their subcontractors' work need to keep an overview of the current construction state. Regulatory issues add up on the requirement to keep track of the current status on site.

The ongoing digitization and the establishment of building information modeling (BIM) technologies in the planning of construction projects help to establish new methods for process optimization. In an ideal implementation of the BIM concept, all semantic data on materials, construction methods, and even the process schedule are connected. On this basis, it is possible to make much more precise estimations about the project costs and its duration. Most importantly, possible deviations from the schedule can be detected early, and the resources can be adapted accordingly.

This technological advancement allows new methods in construction monitoring. In Braun et al. [1], the authors propose a method for automated progress monitoring using photogrammetric point clouds and 4D Building Information Models. The central concept is to use standard camera equipment on construction sites to capture the current construction state by taking pictures of the complete facility under construction at regular intervals. As soon as a sufficient number of images from different points of view are available, a 3D point cloud can be reconstructed with the help of photogrammetric methods. This point cloud represents one particular time-stamp of the construction progress (as-built) and is subsequently matched against the geometry of the BIM (as-planned) on a per-element basis.

Figure 1 shows the C#-based WPF software tool, developed in the scope of this research. The tool visualizes a building information model and all corresponding semantic data. Additionally, the observation results can be

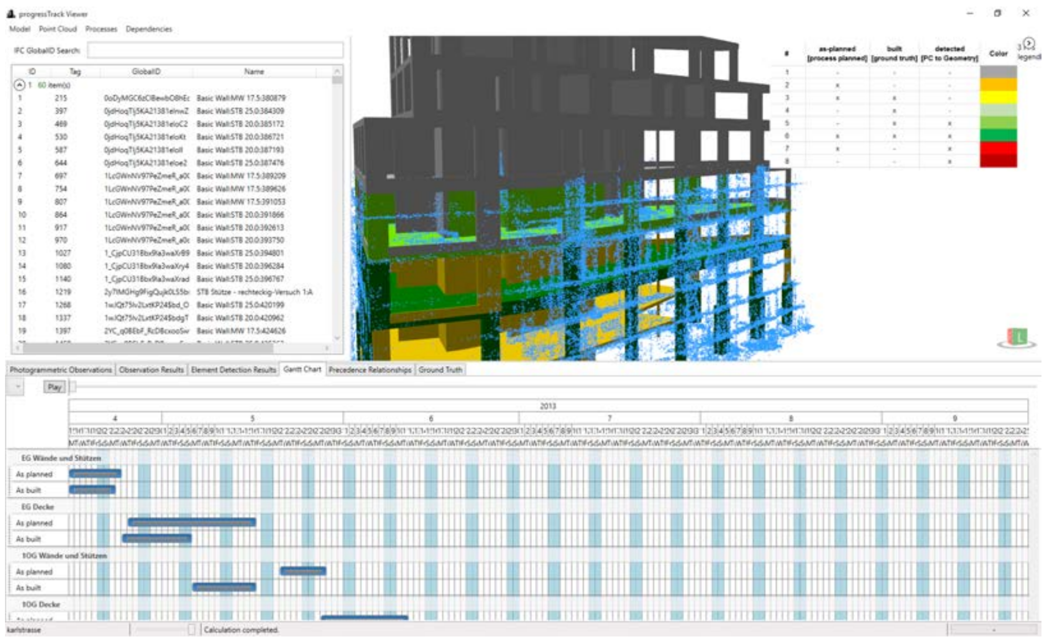


Figure 1: progressTrack: 4D BIM viewer incorporating detection states, process information and point clouds from observations

selected and are supported by the possible overlay of the corresponding point clouds.

The presented approach can be varied in terms of acquisition method (laser scanning, manual acquisition, . . .) and matching methods (as discussed in Section 2 - Related work). However, none of the methods is capable of providing absolute reliability due to occlusions or other boundary conditions. To further improve the reliability of the methods mentioned above, image-based machine learning techniques offer a promising approach. These techniques allow to analyze pictures based on their contents and even mark and classify specific regions of pictures. This new information can further improve the geometric as-planned vs. as-built comparison based on point clouds by increasing the reliability of made assumptions while comparing semantic data from the BIM with classified categories on similar pictures.

Recently, Convolutional Neural Networks (CNN) were introduced in this context [2, 3]. These networks require large training sets to learn similarities of provided data-sets to make assumptions on unknown data. Applications of CNNs range from face-detection in security-related applications to autonomous driving [4]. With respect to automated construction monitoring, these methods can help to detect construction elements on pictures and provide an alternative method for detection in case of low point cloud densities and to improve the overall accuracy of detection [5, 6]. However, data pre-processing and labeling of test-sets for the training of said algorithms is a laborious and time-consuming task since common CNNs require large amounts of labeled data [7].

This paper presents a method to automate the process of construction-site image labeling. The proposed method makes use of available information on image localization from the photogrammetric process as well as information on the presence of individual construction elements from the as-planned vs. as-built comparison by the process described above. The resulting availability of training data provides the basis for applying the trained CNN for image-based object detection on any construction site, in particular, those where a 4D-BIM does not exist or only a limited number of images are taken, and the generation of a point cloud is not possible. However, this paper does not report on these next stages but focuses on the provision of correctly labeled images as an essential first step.

2. Related work

2.1. Automated construction monitoring

Several methods for BIM-based progress monitoring have been developed in recent years [8]. Basic methods make use of minor technical advancements like introducing email and tablet computers into the manual monitoring process. These methods still require manual work, but already contribute to the shift towards digitization. More advanced methods try to track individual building components through radio-frequency identification (RFID) tags or similar methods (for example QR codes).

Current state-of-the-art procedures apply vision-based methods for more reliable element identification. These methods either make direct use of photographs or videos taken on site as input for image recognition techniques or apply laser scanners or photogrammetric methods to create point clouds that hold point-based 3D information and additionally color information.

Bosche and Haas [9], Bosché [10] present a system for as-planned vs. as-built comparisons based on laser-scanning data. The generated point clouds are co-registered with the model using an adapted Iterative-Closest-Point-Algorithm (ICP). Within this system, the as-planned model is converted into a point cloud by simulating the points using the known positions of the laser scanner. For verification, they use the percentage of simulated points, which can be verified by the real laser scan. Turkan et al. [11] use and extend this system for progress tracking using schedule information for estimating the progress in terms of earned value and for detecting secondary objects. Kim et al. [12] detect specific component types using a supervised classification based on Lalonde features derived from the as-built point cloud. An object is regarded as detected if the type matches the type present in the model. As above, this method requires that the model is sampled into a point representation. Zhang and Arditi [13] introduce a measure for deciding four cases (object not in place, point cloud represents a full object or a partially completed object or a different object) based on the relationship of points within the boundaries of the object and the boundaries of the shrunk objects. The authors test their approach in a very simplified artificial environment, which is significantly less challenging than the processing of data acquired on real construction sites.

In comparison with laser scanning, photogrammetric methods are less accurate. However, standard cameras have the advantage that they can be used more flexibly, and their costs are much lower. This leads to the need for other processing strategies when image data is used. Omar and Nehdi [8] give an overview and comparison of image-based approaches for monitoring construction progress. Ibrahim et al. [14] use a single camera approach and compare images taken during a specified period and rasterize them. The change between two time-frames is detected using a spatial-temporal derivative filter. This approach is not directly bound to the geometry of a BIM and therefore cannot identify additional construction elements on site. Kim et al. [15] use a fixed camera and image processing techniques for the detection of new construction elements and the update of the construction schedule. Since many fixed cameras would be necessary to cover a whole construction site, more approaches rely on images from hand-held cameras covering the whole construction site.

For finding the correct scale of the point cloud, stereo-camera systems can be used, as done in [16, 17, 18]. Rashidi et al. [19] propose using a colored cube of known size as a target, which can be automatically measured to determine the scale. Additionally, image-based approaches can be compared with laser-scanning results [20]. The artificial test data is strongly simplified, and the real data experiments are limited to a small part of a construction site. Only relative accuracy measures are given since no scale was introduced to the photogrammetry measurements. Golparvar-Fard et al. [21, 22] use unstructured images of a construction site to create a point cloud. The orientation of the images is computed using a Structure-from-Motion process (SFM). Subsequently, dense point clouds are calculated. For the comparison of as-planned and as-built geometry, the scene is discretized into a voxel grid. The construction progress is determined in a probabilistic approach, in which the threshold parameters for detection are determined by supervised learning. This framework makes it possible to take occlusions into account. This approach relies on the discretization of space as a voxel grid to the size of a few centimeters. In contrast, the approach presented here is based on calculating the deviation between a point cloud and the building model directly and introduces a scoring function for the verification process.

The mentioned approaches provide valuable enhancements for automated construction progress monitoring. However, so far, not all potential benefits from using semantic BIM data are unlocked to their full extent. Also, current research does not present solutions for occluded elements as well as temporary construction elements like scaffolds. These elements cover large parts of construction sites and thus cannot be neglected. The presented approach tries to solve this issue by analyzing the images taken during the SFM process.

2.2. Computer Vision

Computer Vision is a heavily researched topic, that got even more attention through recent advances in autonomous driving and machine learning related topics. Image analysis for construction sites, on the other hand, is a rather new topic. Since one of the key aspects of machine learning is the collection of large data-sets, current approaches focus on data gathering. In the scope of automated progress monitoring, Han and Golparvar-Fard [23] published an approach for labeling based on the commercial service Amazon Turk. Chi and Caldas [24] used first versions of neural networks to detect construction machinery on images, Kropp et al. [25] tried to detect in-door construction elements based on similarities, focusing on radiators. Kim et al. [26] used ML-based techniques for construction progress monitoring. They analyzed images by filtering them to remove noise and uninteresting elements to focus the comparison on relevant construction processes. Other publications mainly focus on defect detection (like for example cracks) in construction images [27].

Current research mainly uses manual labels for computer vision. Additionally, no construction data set is currently covering the whole amount of construction elements. An automated labeling approach could better this lack of data to further improve machine learning methods in this scope of application.

3. Problem statement

Monitoring of construction sites by applying photogrammetric methods has become a common practice. Currently, several companies (for example Pix4D, DroneDeploy) provide commercial solutions for end users that allows to generate 3D meshes and point clouds from UAV-based site observations. All these methods give reasonable solutions for finished construction sites or visible elements of interest.

However, there are still many unsolved problems in monitoring construction sites. Photogrammetric methods are sensitive to low structured surfaces or windows. Because of the used method, each element needs to be visible from multiple (at least two) different points of view. Thus, elements inside of a building cannot be reconstructed as they are not visible from a UAV flying outside of the building. Monitoring inside a building is currently still under heavy research [28] and not available in an automated manner as orientation and observation in such mutable areas like construction sites is hard to tackle. These problems lead to holes or misaligned points in the final point cloud, that hinder accurate and precise detection of building elements. On the other hand, laser scanning requires many acquisition points and takes significantly more time and manual effort for acquisition. Finally, both techniques remain with occlusions for regions that are not visible during construction.

As can be seen in Figure 2, another problem is elements that are occluded by temporary construction elements. Especially scaffolds and formwork elements occlude the view on walls or slabs, making it harder for algorithms to detect the current state of construction progress.

This paper proposes a method that is meant to overcome some of the limitations of the available methods. It contributes to the final goal of exploiting images as an information source for construction state detection, either as additional information in case one of the methods mentioned above is applied, or even as sole and primary information if a 4D BIM does not exist or an insufficient number of images is available for photogrammetric detection. To achieve this, the authors propose to apply CNNs for automated object detection. However, a huge set of correctly labeled images is required for training

Figure 2: Occluded construction elements in generated point cloud caused by scaffolding, formworks, existing elements and missing information during the reconstruction process

the CNN and achieve high precision and low recall. So far, the labeling process had to be performed manually in a laborious and error-prone process. This is why the authors propose to automate this process by making use of the methods they originally developed for construction progress monitoring. In particular, we use image localization from the photogrammetric process as well as information on the presence of individual construction elements from the as-planned vs. as-built comparison. This results in the availability of the required high quality, high volume training data.

4. Automated labeling of images

An essential part of progress monitoring is the detection of an element's status, i.e. to decide whether an element is still under construction (e.g., surrounded by formwork) or finished. Pure point-cloud-to-model matching methods are facing difficulties in this regard as temporary and auxiliary constructions (such as formwork) usually are not included in the BIM model. As proposed in Braun et al. [29], computer vision based methods can help here and significantly improve the reliability of as-planned vs. as-built comparison. The basic idea is to use visual information to decide upon an element's visibility status.

The authors propose the use of machine learning (ML) methods for image-based detection of a construction element's status. However, ML techniques require a large set of labeled images for training. As currently large labeled sets of construction site images or not available, the labeling has to be performed manually in a tedious and time-consuming process. Generating these labels automatically can drastically reduce preparation efforts for training and improving such networks.

The proposed concept of automatic labeling is based on fusing information available from the photogrammetric process (images and relative position of the camera) and the information available from the 4D BIM (object type, object position). Since the BIM and the resulting point cloud are aligned, each BIM element can be projected onto the image initially taken for the photogrammetric process. This allows to precisely identify the region covered by a building element on a picture.

However, there is a significant problem remaining: Information on the actual presence of the element cannot be reliably taken from the 4D as-planned BIM, as execution time very often deviates from the original schedule (which is the underlying rationale for applying progress monitoring). At this point, we benefit from the original point-cloud vs. BIM matching process outlined in Section 1: It provides reliable information about the actual presence of an element in reality and thus also on the captured images.

Consequently, the proposed method for automated labeling of construction elements uses the data of previously monitored construction sites together with the results from the as-planned vs. as-built comparison to generate valid data sets for the training of neural networks.

The proposed workflow is also depicted in Figure 3.

As soon as the training is successfully completed, these networks can be used on any construction site for an image based detection of elements.

The following subsections describe the process and mathematical background for the projection of construction elements into pictures and the labeling procedure using these results.

4.1. Camera positions

In the proposed method, the point cloud is produced using photogrammetric methods. In this process, pictures are taken, for example by UAVs (Unmanned aerial vehicles) from different points of view. These pictures can then be used to generate a 3D point cloud if all elements are visible from a

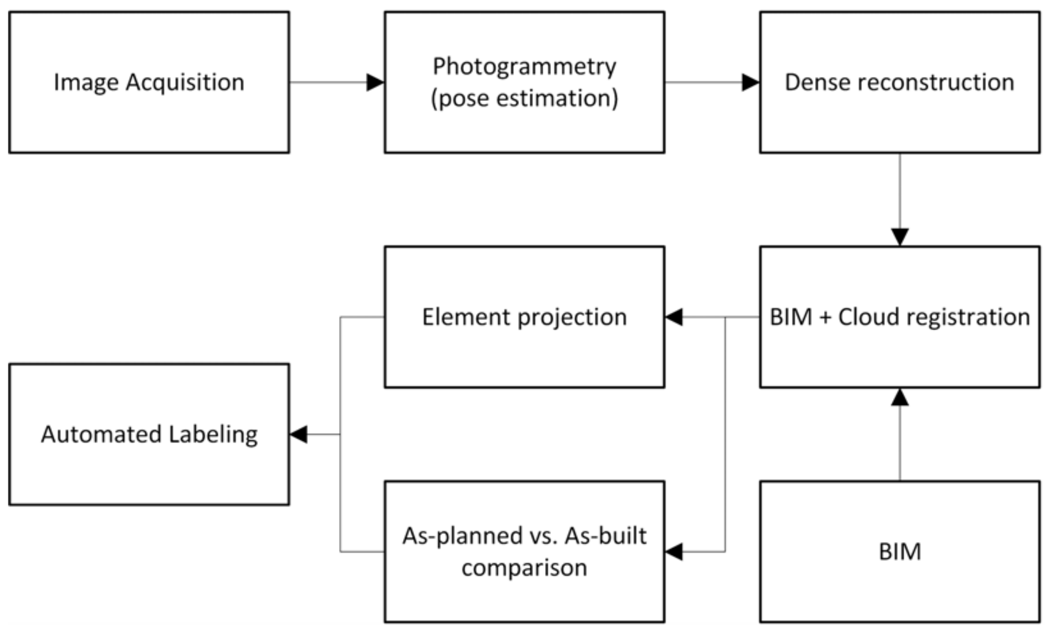


Figure 3: Proposed workflow for the automated labeling toolchain

sufficient amount of viewpoints. During the reconstruction process, the camera positions around the construction site are estimated. This is illustrated in Fig. 4. This estimation is refined during the dense reconstruction and can get more accurate by using geodetic reference points on site.

4.2. 4D process data and as-planned vs. as-built comparison

Building information modeling can be used to combine the geometry of construction elements with semantic data such as material information but also process schedules. In the scope of this research, the corresponding process schedule is connected to all elements, resulting in a fine-grained 4D-BIM model. This allows identifying all elements that are expected to be built at each observation time.

As depicted in Fig. 5, the software tool used in this research is capable of integrating the building information model with process data and construction elements such as scaffolding and formwork.

This data is required to define the sets of elements that are used for the labeling method described in this paper. Since the process schedule may change during construction, it is crucial to update the schedule permanently based on the gathered observation data. Since the as-planned vs. as-built

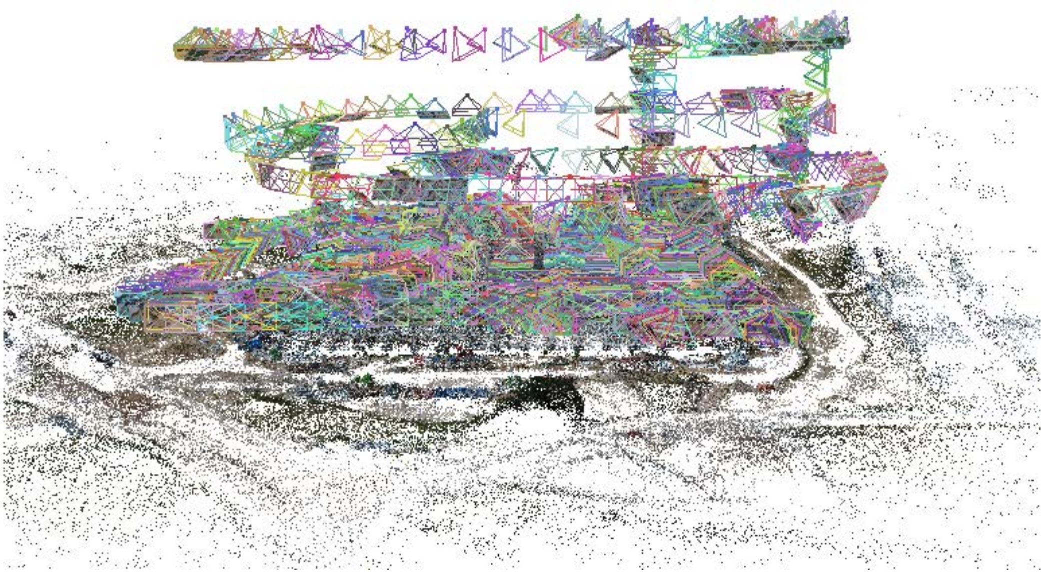


Figure 4: Estimated camera positions during point cloud generation (in this example using VisualSFM [30])

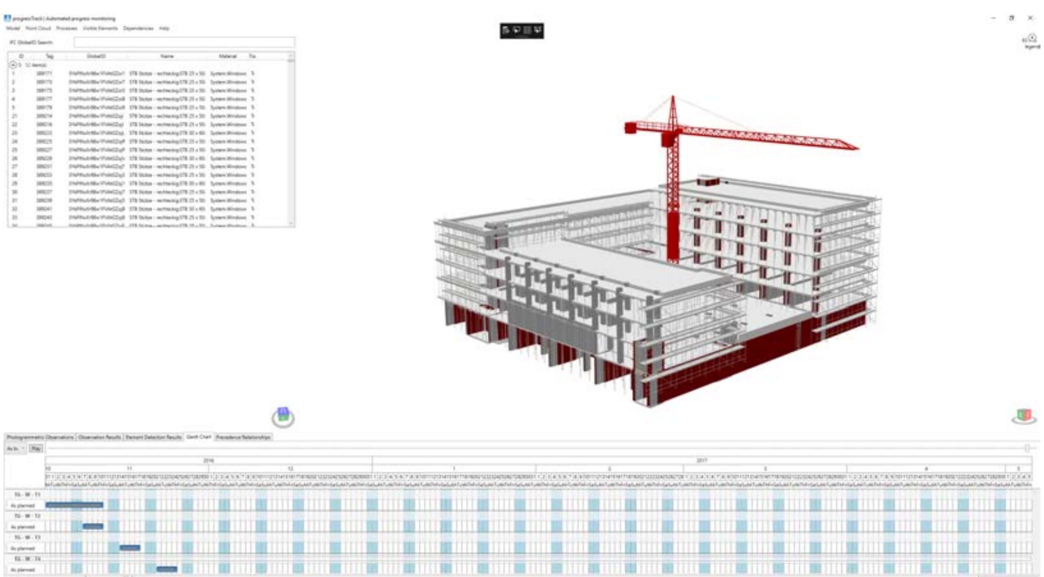


Figure 5: 4D building information model including all additional construction materials like scaffolding and formwork

comparison has already been conducted for the construction sites in this research, the results are available for all construction elements. This information is crucial since the labeling of elements that were not built yet would lead to incorrect labels.

4.3. Projection

Based on the gathered information, it is possible to do a visibility detection by using the camera positions as the point of view, and the process information to define the set of construction elements, that are meant to be built. To achieve this, the building model coordinate system needs to be transformed into the camera coordinate system or vice versa. Several parameters are needed for this transformation.

On the one hand, the intrinsic camera matrix for the distorted images that projects 3D points in the camera coordinate frame to 2D pixel coordinates using the focal lengths (F_x, F_y) and the principal point (x_0, y_0) is required. Additionally, the skew coefficient s_k for the camera is required. This scalar parameter defines the relation between x and y axis. It is zero if the image axes are perpendicular. The matrix K can be described as defined in equation 1.

$$K = \begin{bmatrix} F_x & s_k & x_0 \\ 0 & F_y & y_0 \\ 0 & 0 & 1 \end{bmatrix} \tag{1}$$

The translation of the camera is defined as:

$$T = \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix} \tag{2}$$

Additionally, the rotation matrix for each image as defined in equation 3 is needed.

$$R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \tag{3}$$

Both, translation and rotation can be described in one 3 x 4 matrix:

$$RT = \begin{bmatrix} r_{11} & r_{12} & r_{13} & T_1 \\ r_{21} & r_{22} & r_{23} & T_2 \\ r_{31} & r_{32} & r_{33} & T_3 \end{bmatrix} \tag{4}$$

Using the model coordinates of all triangulated construction elements, it is possible to calculate the projection of each element into the camera coordinate system and therefore overlay the model projection and the corresponding picture taken from the point of observation with equation 5.

$$t = K * RT * p; \tag{5}$$

The resulting 2D coordinates that are rendered into the picture are calculated by using the vector t and getting the x and y coordinates by calculating

$$x = t[0]/t[2] \tag{6}$$

and

$$y = t[1]/t[2] \tag{7}$$

This is done for each point belonging to the triangulated geometry representation of all construction elements.

As visible in Fig. 6 for an analytical column, the projection works as expected and helps to identify the respective construction element in the recorded picture. The mentioned calculations need to include an optional transformation and rotation if the model is geo-referenced and thus the two coordinate systems differ broadly.

4.4. Render model based on camera position

The algorithm introduced in section 4.3 enables the element-wise rendering of all construction elements in the respective coordinate system. To get a rendered image of all visible construction elements, the following steps are carried out:

While all geometric information is available, three problems need to be solved for an accurate rendering of all construction elements:

1. For triangulated elements, only the boundaries are known. However, the whole surface needs to be rendered correctly.
2. The rendered surface needs to be connected to the corresponding element since this information is crucial for a proper visibility analysis

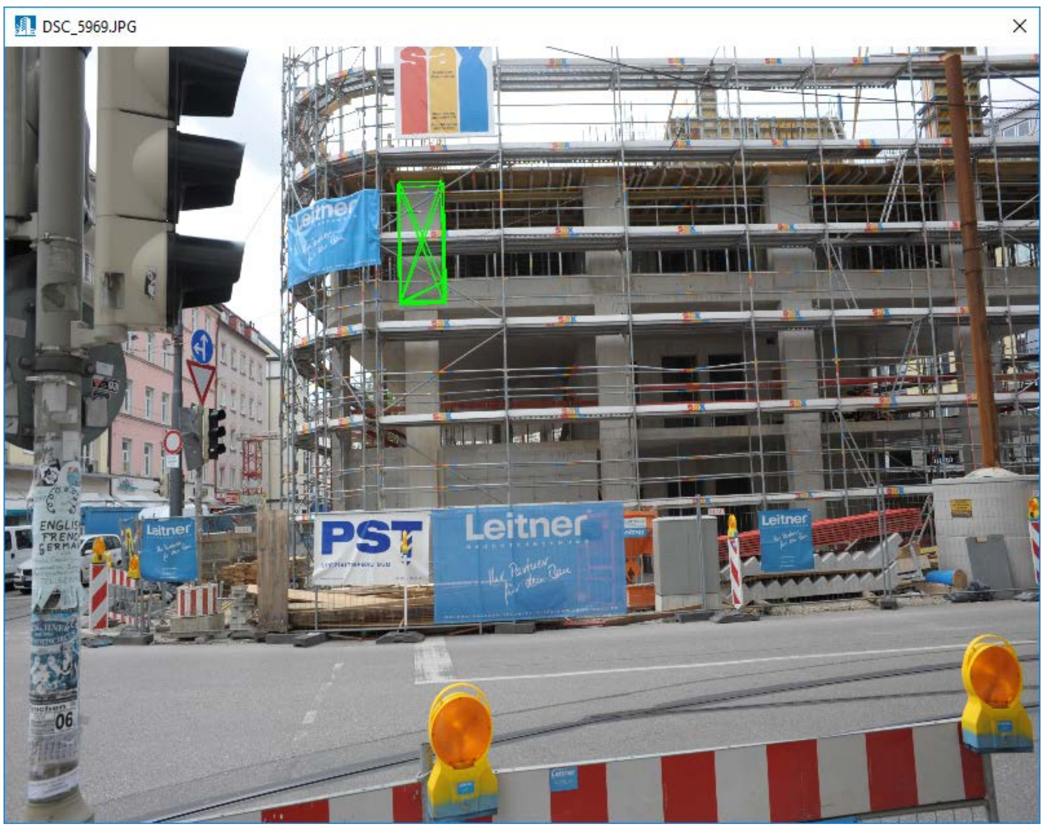


Figure 6: Sample of projected, triangulated column geometry into a corresponding picture

Algorithm 1 Pseudo code for rendering an image of all visible elements

procedure RenderVisibleElements
  $O \leftarrow$ set of all *observations of the construction site*
  $I \leftarrow$ set of all *images of current observation*
  $E \leftarrow$ *set of all construction elements*
  $C \leftarrow$ *set of all coordinates of the triangulated surfaces*
  $d \leftarrow$ *distance of element to corresponding camera position*
  for all O **do**
    for all I **do**
      for all E **do**
        for all C **do**
          if *isvisible*(c) **then** P(x,y,d,color) = projection(c);
  for all *Pixels* **do**
    $p_{min} = min(P(d));$
    $p(x, y) = p(color);$

3. Elements may blend over from the viewpoint in some circumstances. This needs to be addressed to get a correct rendering.

The first issue is solved by applying necessary inside/outside tests for points inside a bounding box around each triangle. This is combined with min/max tests to verify that all points are inside the given coordinate system of the current picture. The second issue is addressed by assigning an individual color in the RGB color range to every construction element. This allows identifying each element after the rendering is finished.

The third issue is solved by applying the Painter's algorithm [31] to each pixel in the given picture. In the given challenge, the distance to the point of view is stored for the current construction element and the color information is replaced in case an element has a smaller distance to the point of view and is also visible in the same pixel of the picture.

The applied algorithms result in a rendering as seen in Fig. 7.

After applying this technique to all observations and all camera positions, a distinct list of all visible construction elements can be generated by iterating over all pixels of each rendered image. The color of each pixel is assigned to a construction element, and since the painters' algorithm is applied, only the element is visible, that has the lowest distance to the point of observation. Therefore, all visible, non-occluded elements can be determined with this

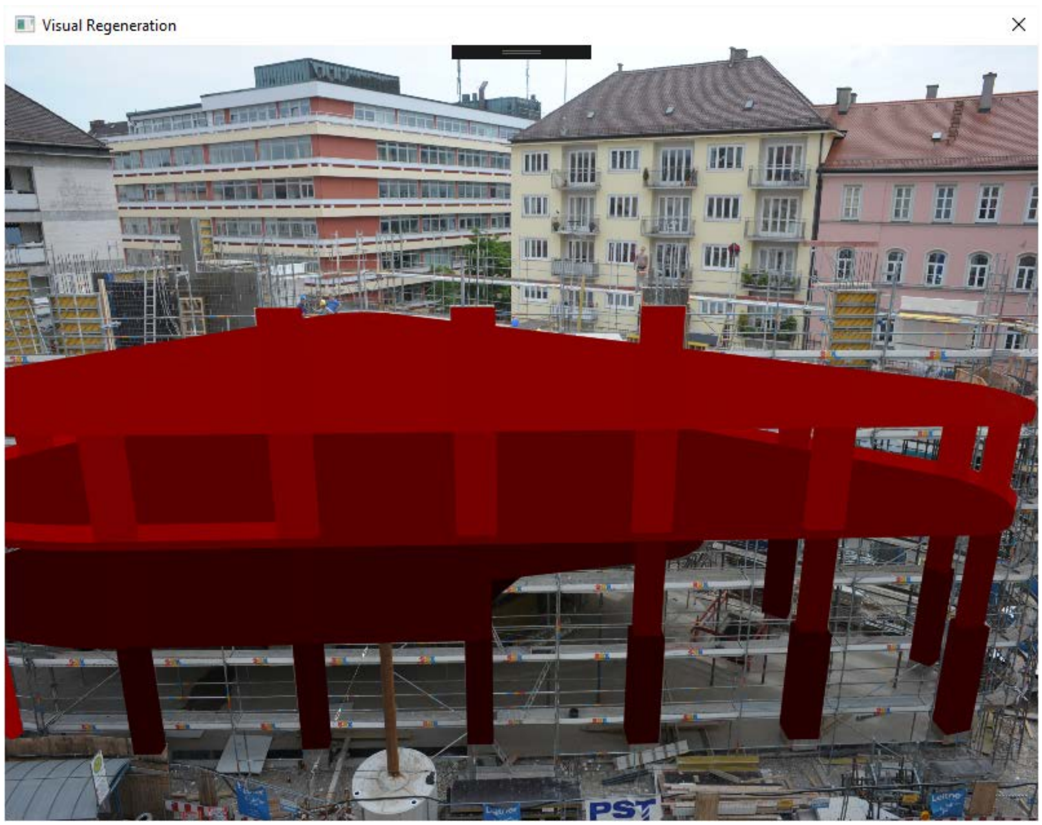


Figure 7: Using projection methodology for model rendering based on the Painter's Algorithm and 4D semantic information

method.

4.5. Generating Labels for Machine Learning

Since machine learning tasks require large training sets for the learning procedure, the labeling and pre-processing of suitable data play a crucial role.

Labeling for ML depends on the desired output of the ML system. A basic ML system for classification is only capable of making general statements on the content of an image and hence only requires a set of images containing the classification category as training input. On the other hand, a system for semantic segmentation can predict the exact location and also the amount of (multiple) elements in one image. Labeling for this class of systems requires detailed convex hull polygons around all instances of elements. Additionally, the category for each label needs to be defined.

The automated labeling process presented here builds on the previously presented projection algorithm and is capable of generating labels for all sorts of ML systems starting from necessary bounding boxes up to detailed convex hulls around individual element instances. Image-based labeling is realized by defining a polygon line around each object and associating a corresponding category with this label. The polygon label can be generated by the above-mentioned projection and fits precisely around the shape of each construction element. The defining element category can be extracted from the semantic information provided by the building information model. Since geometry and semantic data are connected, any additional information can be added to the generated labels.

Besides using the mathematical algorithms for projection, also the results of the visibility analysis are essential. As discussed before and depicted in Figure 8, labeling cannot only rely on all available elements. A prominent but noteworthy factor is the actual presence of the labeled element. The element must have been built to generate an image valid for training or testing. By extracting this information from the as-planned vs. as-built comparison, the set of available elements is reduced to the set of detected elements. In the next step, the set needs to be further reduced to the set of visible elements for each picture.

To sum up the labeling process, the following method is proposed:

The proposed method works for all kinds of label requirements. To illustrate this, Table 1 shows sample labels for a Classification Network (which

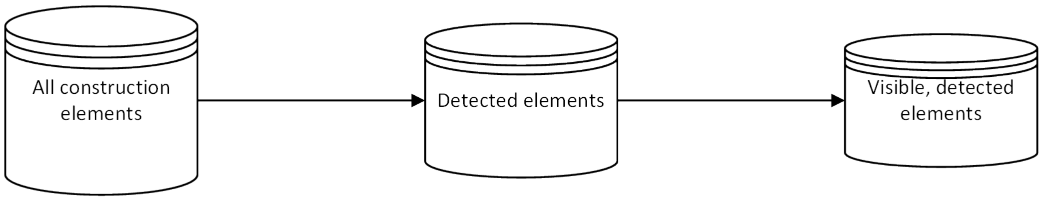


Figure 8: Considered set of elements based on previous results from as-planned vs. as-built comparison. For CV-based methods, visibility plays a crucial role, leading to reduced data sets from construction monitoring.

Algorithm 2 Pseudo code for labeling all visible elements in an image

procedure LabelVisibleElements
  $I \leftarrow$ set of all *images*
  $List < element, List < P >> LabelList \leftarrow$ set of all *labels*
  for all I **do**
    $E \leftarrow$ *set of all construction elements, visible in current picture*
    for all E **do**
      $List < P > ConvexHull = GetConvexHull();$
  $LabelList.Add(E.elementtype, ConvexHull);$

usually requires image snippets with bounding boxes) and semantic segmentation (which usually requires polygon lines and the corresponding images). The sample shown here uses the well known COCO format. For better understanding, a graphic labeled image for semantic segmentation is added, too.

Current best practice in machine learning proposes to split the labeled data-set into a set of training data for the actual training process, a set of validation that does not contain any data from the training set to validate the current training rates. Finally, a set for testing that is not used for training or validation at all is used for checking the overall performance of the neural network without further changing the learning parameters.

Hence, the labeled images are split randomly into the mentioned categories to fulfill this requirement.

5. Case study

To prove the introduced methods, the following case studies were conducted:

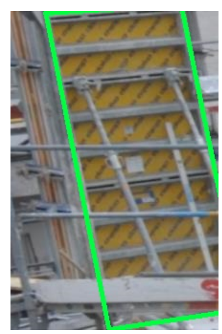
Category	Classification	semantic seg. [JSON coords]	seg. image
column		[[3778, 1230, 3810, 1230, 3834, 1231, 3837, 1230, 3840, 1230, 3854, 983, 3852, 984, 3848, 984, 3791, 985]]	
formwork		[[2662, 1662, 2666, 1682, 2703, 1682, 2704, 1323, 2702, 1319, 2699, 1314, 2663, 1314]]	

Table 1: Sample labels of two categories for different ML use cases

5.1. BIM element projection onto images

The developed methodology has been applied to several construction sites.

As depicted in Fig. 2, most observations lack details at some point and have mostly occluded areas due to the observation methods. In very disadvantageous observations, the detection rate can drop down to 50% of the overall built construction elements. In this case, the detection rate d describes the percentage of elements that the proposed method marked as detected over the ground truth of all elements that were built. The latter set of elements has been acquired manually in order to verify used algorithms. With the help of the presented methods, these rates can be explained since most of the undetected elements were not visible from the observation points. To quantify the efficiency of an algorithm for as-planned vs. as-built detection, it is essential to have a valid ground truth to allow an unbiased evaluation of the used methods. This approach helps to quantify the used methods correctly.

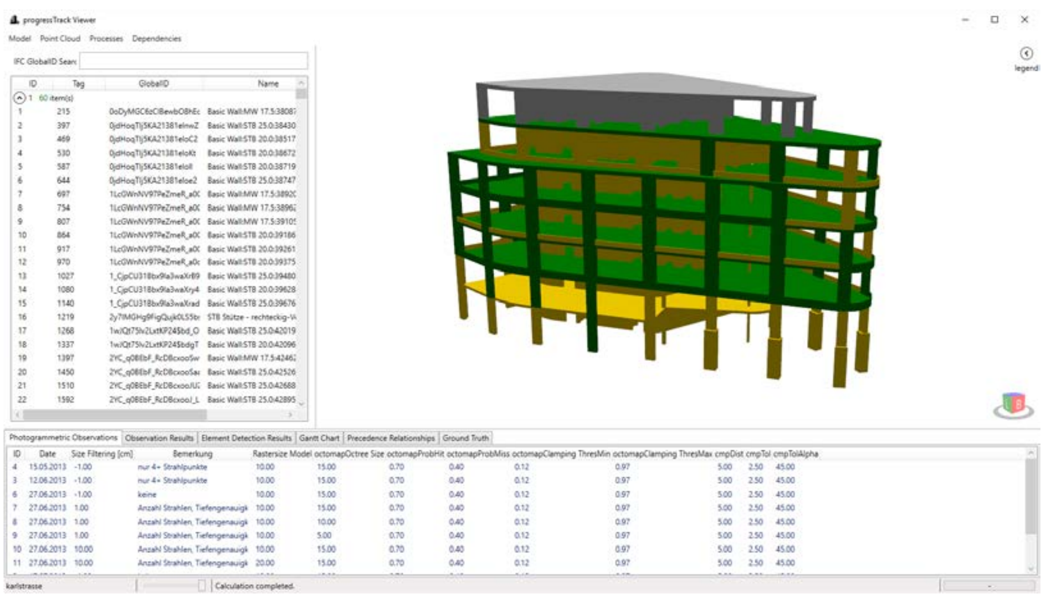


Figure 9: Detected construction elements from one observation. Green elements were successfully detected, yellow elements were not detected but are built.

This concept is illustrated in Fig. 9. The green elements were detected correctly. The yellow elements, however, are built but were not detected.

This is because the inner walls were not visible from a sufficient number of viewpoints. Thus there were not enough points in the corresponding point cloud that allowed to validate the existence of the elements. However, these elements were identified as not visible using the method introduced earlier in this paper.

5.2. Automated labeling and validation

After successfully testing the projection, the actual labeling is performed being the key contribution of this paper.

Many currently used CNNs rely on the COCO Data-set [32]. Facebook's Mask R-CNN [33] has provided promising results for machine learning in previous applications. The network itself also relies on the COCO data format as a basis. Thus, the authors chose this schema as a basis for the generation of the labels. This schema requires a defined structure for all labels, including information about each image (id, width, height, license, date captured), all annotations (id, corresponding image, label category, label polygon, bounding box, ...) as well as the defined categories (in this case for example walls or columns, all represented by individual IDs).

The construction projects on which the developed methods have been applied involve mainly the production of concrete elements. The following construction elements and temporary elements were modeled in the corresponding BIM:

- columns
- walls
- formworks
- slabs
- roofs
- stairs

The proposed methods were tested on observation data from multiple construction sites, resulting in 32,787 labeled construction elements on 1,300 images. The machine used for this test is a Windows 10 system equipped with an Intel Xeon E5-1630 CPU @ 3.70GHz, 16 GB DDR4-RAM, AMD Fire Pro W4100 2GB RAM, and a 10GBit network connection. The entire

Figure 10: Sample sub-set of auto-labeled columns in one picture from a construction site.

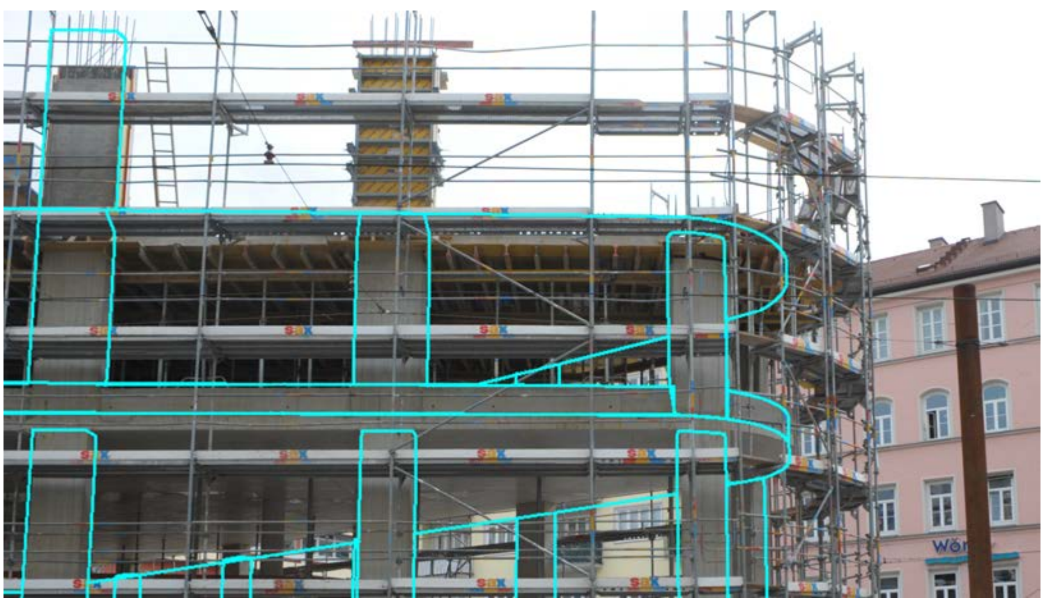


Figure 11: Sample sub-set of auto-labeled columns and walls in another picture from a construction site.

automated labeling process took around 20 minutes, outperforming manual labeling significantly. During this time, all images (close to 9 GB) were downloaded from a NAS (Network attached storage) and randomly added to the training, validation, and test data sets. Additionally, the corresponding label files in JSON format were generated. A sample visualization of the generated labels for one picture is depicted in Figure 10 and 11, showing exported columns with their respective convex hull label around them.

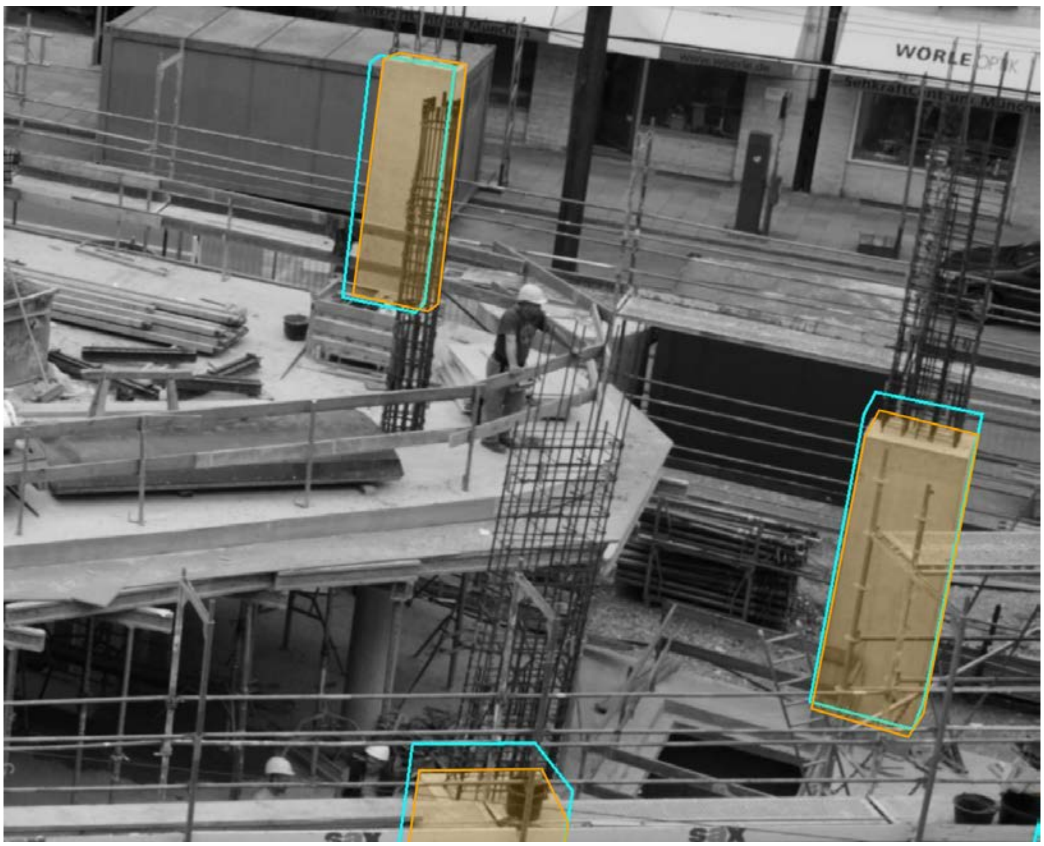


Figure 12: Validation of label correctness with cyan poly-lines representing the automatically generated labels and orange poly-lines representing the manually generated validation set.

The method was validated through human evaluation on all labels for the tested construction sites. The label projection worked without failure for all built construction elements in terms of generating a valid convex hull as the existing elements have been verified against a manually created ground truth.

Since no issues were found in a set of over 32,000 snippets, the projection can be regarded as working correctly. However, as depicted in Fig. 12, the automatically generated labels (cyan poly-lines) have a slight deviation from the actual construction elements.

This deviation can have multiple reasons:

- errors in pose estimation during Structure-from-Motion
- large scale deviations when using real-world coordinates
- construction inaccuracies
- modeling inaccuracies

Since all elements were validated in the as-planned vs. as-built comparison, allowing for only very minor construction inaccuracies, construction inaccuracies can be disregarded in this research. Otherwise, the element would not have been classified as "built" and would not have been labeled at all. Thus, the deviations, in this case, are minor and arise from an aggregation of the mentioned reasons. To quantify the introduced error, a set of 1.000 elements have been labeled manually and tested against the automatically created labels. The labels were then compared pixel-wise. The overall accuracy of the automated system was measured by calculating the overlapping area I of the resulting labels of both labeling methods over the manually labeled area:

$$p_o = I/A_{manuallabel} \tag{8}$$

with

$$I = A_{autolabel} \cap A_{manuallabel} \tag{9}$$

The resulting accuracy p_o had an average of 91.7% overlap over all checked labels, constantly lying within the bounds of 85% and 97%. The overlap rates give promising results and make the labels usable for machine learning tasks. Rates could be further improved by taking more pictures for the Structure-from-Motion process and enhancing the resulting camera pose estimation.

6. Discussion

For improving the reliability of construction progress monitoring, this paper introduces a novel concept for automating the labeling process of construction site images. It is based on fusing information available from the photogrammetric process (images and relative position of the camera) and the information available from the 4D BIM (object type, object position). Since the BIM and the resulting point cloud are aligned, a digital element can be projected onto the image, initially taken for the photogrammetric process. Also, matching the point cloud and the BIM allows to make sure that only images are considered where the elements under consideration exist in reality.

From the projected BIM elements, it is possible to automatically connect the covered image segments with the semantic information provided by the Building Information Model. Since the introduced as-planned vs. as-built comparison also offers valuable information on the presence of all elements, the labels can be further refined regarding possible occlusions. As a valid label should only be applied to an at least partially visible element, the gathered knowledge from the previously applied as-planned vs. as-built comparison makes this automated approach even more accurate. Since the comparisons' resulting elements are built at the correct positions, the labels are also correct. On the downside, only elements that were built as-planned can be labeled.

The sample-based validation showed over 91% pixel-wise accuracy of the automated procedure when tested against manual labeling procedures. A previously tested, manual labeling approach took over 100 working hours to accurately label only one category of elements. Labeling and generating the corresponding images folders for this case study took around 20 minutes, including downloading of 9 GB of pictures from a remote NAS folder which takes over 90% of the time used. Additionally, several studies show that manual labeling is also introducing a range of errors due to missed elements or inaccurately labeled elements. As the correct identification of construction elements also requires technical personnel [34], labeling is hugely cost intensive and danger of bore-out to this group of workers due to the repetitive work.

The construction sites used for this process are located in Germany and apply in-situ concrete pouring as the primary construction methodology. Consequently, the resulting labels and especially the trained network, will only be able to detect construction elements from this domain of manufacturing. However, the presented approach can be easily extended by also including construction sites from other countries or other construction techniques.

Future steps of this research will focus on creating a CNN for detecting the most important construction elements on construction sites. The final objective is to enable a utterly image-based construction monitoring process in the future.

Acknowledgments

We thank the Leibniz Supercomputing Centre (LRZ) of the Bavarian Academy of Sciences and Humanities (BAdW) for the support and provision of computing infrastructure essential to this publication. We thank the German Research Foundation (DFG) for funding the initial phases of the project.

Additionally, we would like to thank all construction companies, providing data and/or support to the research conducted in this field, including: Leitner GmbH & Co Bauunternehmung KG, Kuehn Malvezzi Architects, Staatliches Bauamt Muenchen, Baugesellschaft Brunner+Co, BKL (Baukranlogistik GmbH), Geiger Gruppe, Baureferat H5, Landeshauptstadt Muenchen, Baugesellschaft Mickan mbH & Co KG, h4a Architekten, Wenzel + Wenzel and Stadtvermessungsamt Muenchen.

References

[1] A. Braun, S. Tuttas, A. Borrmann, U. Stilla, Automated Progress Monitoring Based on Photogrammetric Point Clouds and Precedence Relationship Graphs, Proceedings of the 32nd International Symposium on Automation and Robotics in Construction and Mining (ISARC 2015) (2017) 274–280. doi:10.22260/isarc2015/0034.

[2] R. Girshick, J. Donahue, T. Darrell, J. Malik, Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation, in: 2014 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2014, pp. 580–587. URL: http://ieeexplore.ieee.org/document/6909475/. doi:10.1109/CVPR.2014.81.

[3] K. He, X. Zhang, S. Ren, J. Sun, Deep Residual Learning for Image Recognition, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2016, pp. 770–778. URL: http://ieeexplore.ieee.org/document/7780459/. doi:10.1109/CVPR.2016.90.

[4] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You Only Look Once: Unified, Real-Time Object Detection, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2016, pp. 779–788. URL: http://ieeexplore.ieee.org/document/7780460/. doi:10.1109/CVPR.2016.91.

[5] A. Dimitrov, M. Golparvar-Fard, Vision-based material recognition for automated monitoring of construction progress and generating building information modeling from unordered site image collections, Advanced Engineering Informatics 28 (2014) 37–49. doi:10.1016/j.aei.2013.11.002.

[6] I. Brilakis, L. Soibelman, Y. Shinagawa, Material-Based Construction Site Image Retrieval, Journal of Computing in Civil Engineering 19 (2005) 341–355. doi:10.1061/(asce)0887-3801(2005)19:4(341).

[7] J. J. Lin, K. K. Han, M. Golparvar-Fard, A Framework for Model-Driven Acquisition and Analytics of Visual Data Using UAVs for Automated Construction Progress Monitoring, in: Computing in Civil Engineering 2015, American Society of Civil Engineers, Reston, VA, 2015, pp. 156–164. URL: http://ascelibrary.org/doi/10.1061/9780784479247.020. doi:10.1061/9780784479247.020.

[8] T. Omar, M. L. Nehdi, Data acquisition technologies for construction progress tracking, Automation in Construction 70 (2016) 143–155. doi:10.1016/j.autcon.2016.06.016.

[9] F. Bosche, C. T. Haas, Automated retrieval of 3D CAD model objects in construction range images, Automation in Construction 17 (2008) 499–512. doi:10.1016/j.autcon.2007.09.001.

[10] F. Bosché, Plane-based registration of construction laser scans with 3D/4D building models, Advanced Engineering Informatics 26 (2012) 90–102. doi:10.1016/j.aei.2011.08.009.

[11] Y. Turkan, F. Bosché, C. T. Haas, R. Haas, Automated progress tracking using 4D schedule and 3D sensing technologies, Automation in Construction 22 (2012) 414–421. doi:10.1016/j.autcon.2011.10.003.

[12] C. C. Kim, H. Son, C. C. Kim, Fully automated registration of 3D data to a 3D CAD model for project progress monitoring, Automation in Construction 35 (2013) 587–594. doi:10.1016/j.autcon.2013.01.005.

[13] C. Zhang, D. Arditi, Automated progress control using laser scanning technology, Automation in Construction 36 (2013) 108–116. doi:10.1016/j.autcon.2013.08.012.

[14] Y. Ibrahim, T. Lukins, X. Zhang, E. Trucco, a. Kaka, Towards automated progress assessment of workpackage components in construction projects using computer vision, Advanced Engineering Informatics 23 (2009) 93–103. doi:10.1016/j.aei.2008.07.002.

[15] C. C. Kim, H. Son, C. C. Kim, Automated construction progress measurement using a 4D building information model and 3D data, Automation in Construction 31 (2013) 75–82. doi:10.1016/j.autcon.2012.11.041.

[16] H. Son, C. Kim, 3D structural component recognition and modeling method using color and 3D data for construction progress monitoring, Automation in Construction 19 (2010) 844–854. doi:10.1016/j.autcon.2010.03.003.

[17] I. Brilakis, H. Fathi, A. Rashidi, Progressive 3D reconstruction of infrastructure with videogrammetry, Automation in Construction 20 (2011) 884–895. doi:10.1016/j.autcon.2011.03.005.

[18] I. Brilakis, S. German, Z. Zhu, Visual Pattern Recognition Models for Remote Sensing of Civil Infrastructure, Journal of Computing in Civil Engineering 25 (2011) 388–393. doi:10.1061/(ASCE)CP.1943-5487.0000104.

[19] A. Rashidi, I. Brilakis, P. Vela, Generating Absolute-Scale Point Cloud Data of Built Infrastructure Scenes Using a Monocular Camera Setting, Journal of Computing in Civil Engineering 29 (2015) 04014089. doi:10.1061/(ASCE)CP.1943-5487.0000414.

[20] M. Golparvar-Fard, M. Asce, F. Peña-Mora, S. Savarese, M. Asce, F. Peña-Mora, S. Savarese, Integrated Sequential As-Built and As-Planned Representation with Tools in Support of Decision-Making Tasks in the AEC/FM Industry, Journal of Construction Engineering and Management 137 (2011) 1099–1116. doi:10.1061/(ASCE)CO.1943-7862.0000371.

[21] M. Golparvar-Fard, F. Pena-Mora, S. Savarese, Monitoring changes of 3D building elements from unordered photo collections, Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on (2011) 249–256. doi:10.1109/ICCVW.2011.6130250.

[22] M. Golparvar-Fard, F. Peña-Mora, S. Savarese, Automated Progress Monitoring Using Unordered Daily Construction Photographs and IFC-Based Building Information Models, Journal of Computing in Civil Engineering 29 (2015) 04014025. doi:10.1061/(ASCE)CP.1943-5487.0000205.

[23] K. Han, M. Golparvar-Fard, Crowdsourcing BIM-guided collection of construction material library from site photologs, Visualization in Engineering 5 (2017) 14. doi:10.1186/s40327-017-0052-3.

[24] S. Chi, C. H. Caldas, Automated Object Identification Using Optical Video Cameras on Construction Sites, Computer-Aided Civil and Infrastructure Engineering 26 (2011) 368–380. doi:10.1111/j.1467-8667.2010.00690.x.

[25] C. Kropp, C. Koch, M. König, Interior construction state recognition with 4D BIM registered image sequences, Automation in Construction 86 (2018) 11–32. doi:10.1016/j.autcon.2017.10.027.

[26] C. Kim, B. Kim, H. Kim, 4D CAD model updating using image processing-based construction progress monitoring, Automation in Construction 35 (2013) 44–52. doi:10.1016/j.autcon.2013.03.005.

[27] B. Akinci, F. Boukamp, C. Gordon, D. Huber, C. Lyons, K. Park, A formalism for utilization of sensor systems and integrated project models for active construction quality control, Automation in Construction 15 (2006) 124–138. doi:10.1016/j.autcon.2005.01.008.

[28] C. Kropp, C. Koch, M. König, Drywall State Detection in Image Data for Automatic Indoor Progress Monitoring, in: Computing in Civil and Building Engineering (2014), November 2015, American Society of Civil Engineers, Reston, VA, 2014, pp. 347–354. URL: http://ascelibrary.org/doi/10.1061/9780784413616.044. doi:10.1061/9780784413616.044.

[29] A. Braun, S. Tuttas, U. Stilla, A. Borrmann, Process- and Computer Vision-based Detection of As-Built Components on Construction Sites, in: 2018 Proceedings of the 35th ISARC, Berlin, Germany, 2018, p. 7. doi:10.22260/ISARC2018/0091.

[30] C. Wu, Towards linear-time incremental structure from motion, in: Proceedings - 2013 International Conference on 3D Vision, 3DV 2013, IEEE, 2013, pp. 127–134. URL: http://ieeexplore.ieee.org/articleDetails.jsp?arnumber=6599068. doi:10.1109/3DV.2013.25.

[31] T. T. Elvins, A survey of algorithms for volume visualization, ACM SIGGRAPH Computer Graphics 26 (2005) 194–201. doi:10.1145/142413.142427.

[32] T.-Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick, P. Dollár, Microsoft COCO: Common Objects in Context, Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (2014) 3686–3693. doi:10.1109/CVPR.2014.471.

[33] K. He, G. Gkioxari, P. Dollar, R. Girshick, Mask R-CNN, Proceedings of the IEEE International Conference on Computer Vision 2017-Octob (2017) 2980–2988. doi:10.1109/ICCV.2017.322.

[34] K. K. Han, M. Golparvar-Fard, Potential of big visual data and building information modeling for construction performance analytics: An exploratory study, Automation in Construction 73 (2017) 184–198. doi:10.1016/j.autcon.2016.11.004.