

AUTOMATIC INDOOR CONSTRUCTION PROGRESS MONITORING: CHALLENGES AND SOLUTION

Inshu Chauhan¹ and Olli Seppänen¹

¹Aalto University, Helsinki, Finland

Abstract

Indoor construction progress monitoring has challenges like occlusion, light variation, dynamic environment which makes its automation different from outdoor construction progress monitoring. AI/deep learning approaches can help overcome these challenges but using them for indoor construction monitoring raises some issues like lack of annotated image data for construction works. Transfer learning provides the initial solution to the AI related challenges. In our research we use this state-of-the-art method on construction site data and detect as-built stages of a drywall construction. The results are promising with accurate prediction of 3 stages of the drywall process.

Introduction

Construction industry is a core industry and backbone of a nation's economy, yet it suffers from delays and wastage on account of lack of streamlined process. The management of construction process requires progress monitoring, which determines the status of each task in each location. Progress data are required for key decisions of a construction project including work directives to construction crews and progress payments to contractors (Kenley & Seppänen, 2009). Lack of accurate progress data can lead to significant wasted effort for the crews (Seppänen & Görsch, 2022).

The general method of construction progress monitoring consists of manual observations and reporting the data in weekly contractor meetings. The manual observations are subjective as the reported data could be incomplete and not clearly defined (Zhao et al., 2021). To solve the problems with manual data collection, automatic construction progress monitoring is desired. Automatic progress monitoring stores data in digital format and can be used as part of digital twins in construction stage. Digital twins for construction not only include as-designed models and processes but also as-built model and processes during construction (Sacks et al., 2020). Digital twin for as-built process requires interpretation of as-built process data collected from the construction site. Visual data from site provides scope for better and objective interpretation of as-built process data.

Yet, automatic interpretation of collected data poses a challenge because of construction scene scenario. The construction scene is dynamic, immersive and crowded which renders the interpretation of visual data collected

complex (Fathi et al., 2015). The geometries and materials change frequently while construction progresses (Paneru & Jeelani, 2021). Also, there are variations in scene like light variations, presence of dust particles & reflective objects which may affect the accuracy of the data collected on site and consequently the interpretation (Mirzaei et al., 2022). These challenges require different algorithms for pre-processing the images and for feature extraction as presented in their pioneering work by Hamledari et al. (2017) and Kropp et al. (2014) on indoor construction element detection. The research demonstrates promising results, but these algorithms have the inherent drawback that they are specifically tailored to progress detection in certain scenarios. To extend these methods to different construction sites or visual conditions may not produce accurate results. Also, using the same algorithms for new images may require heavy pre-processing to bring the visual conditions to a level where the algorithms could work fine on a new set of data. Thus, there is a need to explore the use of AI and machine learning for construction as-built data interpretation which has been less explored in construction domain.

For our research case, we focus on drywall construction progress monitoring by using Deep learning-based data interpretation. Drywall is chosen as it is an important assembly of indoor construction. BIM (Building Information Models) typically does not include separate elements for each stage of drywall process, so geometry-based methods are not sufficient. This research tries to classify the stage of dry wall in as-built data by utilizing the capability of 'You look only once' (YOLO) object detection algorithm. The challenges which come up while using deep learning in the context of indoor construction progress monitoring are discussed. The resolution of challenges like lack of extensive annotated construction data overcoming the conditions at construction site is an interesting research question which we explore in this paper.

Background

Artificial intelligence and machine learning (AIML) algorithms like Random Forests, SVM and deep learning algorithms like CNN have become popular tools in visual scene interpretation for various applications like autonomous driving, urban scene classification, video surveillance etc (Paneru & Jeelani, 2021). The computer vision (CV) for these applications has advanced to a level of great accuracy. For construction progress monitoring, attempts have been made to automate the progress

assessment based on work package assessment & CV (Braun et al., 2018), by one-to-one comparison between as-built and as-designed data (Yang et al., 2015, Roh et al., 2011, Hamledari et al., 2017). In most works, the use of CV is limited to object classification and comparison between objects is done visually in as-built and as-designed data. (Golparvar et al., 2011, Masood et al., 2020). The approach of comparing as-built to as-designed may be useful for reporting the percentage of work being completed but does not address instances where a single element of construction goes through different stages. This information can be used for optimizing time and resources on construction site, which requires modelling of individual element using object detection and monitoring them using AI and CV.

Some works also demonstrate appearance-based methods to identify changes on construction site based on either geometries or material (Han et al., 2015, Dimitrov and Golparvar, 2014, Son & Kim, 2010, Kevin et al., 2018). Kevin et al. (2018) were successful in capturing element wise progress based on geometries of elements but they did not use a machine learning based approach. In context of construction progress monitoring there is a need to find an efficient and automatic way to align as-built data with as-designed data and to develop algorithms which either do not need a large amount of training data or can be augmented with the help of limited number of labeled classes (Seong et al., 2017). In addition, there have been attempts to recognize some individual objects like rebar covers using specific machine learning algorithms which are particularly tailored to this application (Cuypers et al., 2021). Due to unique characteristics of construction sites, there is also a need to build a comprehensive database for training neural networks which can address variations and dynamics on sites. (Paneru & Jeelani, 2021). Occlusions have been tackled by Xin et al. (2019) by using Fermat paths of light between a known visible scene and an unknown object not in the line of sight of a transient camera to create a prediction of hidden surfaces. However, occlusion and limited visibility issues have not been resolved completely and continue to be a challenge in using computer vision for construction.

There are two prominent research work related to dry wall detection by Kropp et al., (2014) and Hamledari et al. (2017). Both research works utilize traditional image processing i.e. color and texture based object recognition. The first one only identifies 3 stages of dry wall i.e. paneling, plastering and painting using support vector machines. The latter framework can recognize framing, insulation, installed drywall, plastered drywall, and painted partition stages of the dry wall. The work is very promising but has localization errors. The algorithm is specifically tailored to a set of images and using the methods on other construction site images may require heavy pre-processing. Another recent research work by Ekanayake et al. (2022) employs similar approach, but their detection is limited to framing, insulation and drywall installation. We aim to detect all stages of the dry

wall as these phases convey progress at the level which can be important for tracking resources and manpower at the construction site.

The following challenges can be summarized from the literature before AIML can be used impeccably for construction progress monitoring:

1. Due to dynamic site conditions, occlusion (Rebolj et al., 2008) may happen common for both images and point cloud (as-built data).
2. Lack of any algorithms which are trained on construction site objects or large construction site datasets.
3. Lack of fully annotated dataset (Fang et al., 2019) because of new objects, which can appear on construction site.

Moreover, the role of semantics and ontologies in integration with CV needs to be explored for change detection that can provide a crucial clue for construction progress monitoring. Choice of AIML algorithms which do not show what is happening inside these algorithms also pose a challenge for transparent solution development.

Methodology

The initial step in automatic progress monitoring is detection of objects on the construction sites. The detection of objects can be achieved by deriving features by using traditional image processing but that is not universal and application specific. For a more holistic approach, a generalizable solution is more desirable which can be achieved through deep learning methods. Supervised deep learning methods require labeled or annotated data for the model training. Therefore, in this paper we focus on two problems: (1) the lack of domain-specific annotated data and (2) the absence of deep learning models tailored to the construction progress monitoring.

The experiments are done on dry wall because it has distinguishable stages of construction which are typically not included in BIM models. Therefore, just comparing BIM model to images is not sufficient. Our aim is to have a holistic approach which can detect all the as-built stages of dry wall to improve situational awareness on construction sites. Therefore, we choose the stages of drywall so that it covers all the prominent steps in its installation which are as follows, Stage 0: No dry wall; Stage 1: Installation of studs (figure 1); Stage 2: Gypsum Paneling; Stage 3: Electrical and plumbing works (figure 2); Stage 4: Insulation work; Stage 5: Wall-closed/only paneled side visible without studs; Stage 6: Plastering of the wall (figure 3); Stage 7: Painting of the wall.

We formulate our problem as a computer vision task, specifically object detection, which amounts to detection and location of objects of interest in an image. We use deep learning approach in combination with transfer

learning to train a model for object detection (Pan et al., 2009). Transfer learning methods allow leveraging already obtained knowledge for solving different, but related task. As an example, first layers of a convolutional neural network pre-trained on CIFAR image dataset (Krizhevsky, 2009) can be reused for classification problem on a custom image dataset. Transfer learning can be viewed as a form of model regularization. Thus, it is particularly beneficial for reducing overfitting of large neural networks to small datasets. In addition, transfer learning reduces training time and saves computational resources, compared to training deep learning models from scratch. Given these arguments, applying transfer learning approach to construction domain seems very promising.



Figure 1: Stage 1-Installation of Studs for the dry wall



Figure 2: Stage 3-Electrical and plumbing works



Figure 3: Stage 6: Plastering of the wall

In the following experiments we use pre-trained YOLO v7 deep neural network model commonly used for fast, real-time object detection. It was first introduced by Redmon et. al. (2016) to modify object detection as a regression problem to spatially separated bounding boxes

and associated class probabilities. The advantage of using YOLO is that the algorithm utilizes a single neural network to predict objects in images making it an optimized object detection pipeline.

YOLO v7 (Wang et. al, 2022) is the latest version of YOLO model and is originally trained on COCO dataset which has images of 91 objects of common occurrence like car, hat, umbrella, dogs, suitcase, tennis racket, cup, cake, door etc. There are no existing neural network models in public domain which are trained on extensive construction site data. Thus, the first task is to create an annotated dataset for indoor construction works.

Data Collection and Annotation

The data was initially collected in the form of 360 degree videos on two construction sites. Drywall snapshots were taken from these videos to generate images for each stage of the dry wall. We have labeled 100-150 images per each stage, with the assumption that this is a minimal sample size needed for fine-tuning pre-trained neural network. The images were annotated using bounding box with labels as YOLO accepts annotation in the form of bounding boxes. Sample images of annotation are shown in the figures 1, 2 & 3.

The images were annotated within a software called Image Annotation Lab. Before choosing Image Annotation Lab as the labelling software, comparisons were made with VGG annotator, MakeSense, RoboFlow and Supervisely based on parameters such as easy GUI, price and output formats provided. Most of the stages had the target number of labelled images but some stages could not be captured very extensively like insulation because the stage is completed too fast during construction. This class imbalance has to be taken into account during training of the YOLO model and interpreting results.

Next, we used pre-trained the YOLO v7 neural network on the custom dataset consisting of drywall stages and respective bounding boxes. Figure 4 shows the original YOLO v1 architecture which consists of a CNN backbone for feature learning and extraction and a head for prediction of class probabilities, bounding box objectness score (confidence score as described in Equation 1) and bounding boxes coordinates. We applied transfer learning, specifically fine-tuning of the pre-trained network to train YOLO v7 on our custom dataset.

$$Pr(Class_i|Object) * Pr(Object) * IOU_{pred}^{truth} = Pr(Class_i) * IOU_{pred}^{truth} \quad (1)$$

where,

$Pr(Class_i|Object)$ represents the conditional probability of the predicted class i given the presence of an object in the bounding box.

$Pr(Object)$ represents the marginal probability of the presence of an object in the bounding box. It is the probability that there is an object in the bounding box, regardless of what the predicted class is.

IOU_{pred}^{truth} represents the intersection over union (IOU) between the predicted bounding box and the ground-truth bounding box. The IOU is a measure of the overlap between the two boxes, with a value of 1 indicating a perfect overlap and a value of 0 indicating no overlap. $Pr(Class_i)$ represents the marginal probability of the predicted class i . It is the probability that the predicted class is i , regardless of whether an object is present in the bounding box.

The head i.e. classifier is modified as per the number of classes or the type of prediction. In our case it is a multi-label, multi-class prediction, so head of the YOLO is modified as per the requirements. Since construction site images are a new type of data for YOLO, we try to customize the parameters of the neural network. This implies that weights were updated only for the last convolutional layers before the classification head (fine-tuning) with a small learning rate. The details of images per class are shown in table 1. The number of images for Stage 4 i.e. Insulation and Stage 7 i.e. Painting of the dry wall were too low in the collected data set. Therefore, they are not included in this phase of object detection and progress monitoring.

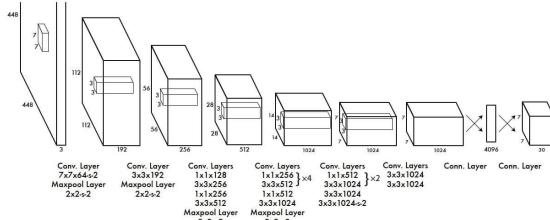


Figure 4: YOLO architecture

Table 1: Image of each stage of dry wall

Name of Stage	No. of Images
Stage 1 Studs	86
Stage 2 Gypsum Paneling	171
Stage 3 Electrical	111
Stage 5 Gypsum Panel Visible Side	199
Stage 6 Gypsum Plaster	130

Results

The experiment was configured in Google Collaboratory (Google colab) by downloading the Yolov7 model from GitHub. (Wang et. al, 2022). The image dataset was divided into 2 categories, the first one containing 80 % of data to be used for training and the rest 20% to be used for validation. Yolo7.pt containing weights based on MS COCO dataset was downloaded and put into the Yolov7 directory to be used as for transfer learning. In the coco.yaml file the location of training and validation set were defined and classes were modified according to our requirements based on 6 classes of drywall. The entire

folder was then uploaded to google drive to be used with Google Colab. The default training parameters were used for the training images, the image dataset was standardized with no rotation, translation needed. Training was done on the Tensor processing unit (TPU) provided by Google colab with batch size of 8 for total of 50 epochs.

After the training was completed, the resulting weights were stored in the best.pt file in the training folder. These weights were then copied to the main directory to be used for object detection. After training, the updated model is tested on the validations set and the recall, the mean average precision (mAP) and average loss at an intersection over union (IoU) of 0.5 is observed from each stage of drywall. The F1 score is not reported as the dataset is small, and the score does work well with class imbalance.

Table 2 summarizes the precision, recall and mAP for all the given stages. Recall and mAP was highest for stage 2 i.e. Gypsum paneling followed by stage 5 and 6. This can also be observed from the figures 5 and 6 where the model has predicted gypsum paneling visible side with .80 confidence score and Gypsum plaster with .53 confidence score. The model predicts the right class with a confidence score (For example a .80 confidence score will mean that the model is 80 % confident that the object detected by the bounding box is present in the image) depending on the similarity of features derived from training images and test images. Lowest recall & mAP is observed for stage 1 i.e studs and stage 3 i.e. electrical and plumbing works, so the model in its current form is not able to detect these stages. An example of this is shown in figure 7 where the model predicts a stage as stage 2 i.e. Gypsum paneling but there are some electrical wires visible which implies it should be Stage 3 Electric work stage.

Table 2: Summary of results on validation set

Name of Stage	Precision	Recall	mAP@0.5
Studs	1	0	0.00195
Gypsum Paneling	0.465	0.855	0.657
Electric Works	0.11	0.0769	0.0648
Gypsum Plaster Visible side	0.376	0.607	0.478
Gypsum Plaster	0.255	0.5	0.27

Discussion

The model in its current form is able to detect 3 stages well but is unable to detect the first and the third stage at all. This may happen due to several factors which influenced the model training and feature extraction from the input images. The first reason could be the number of input images given to neural network for each stage. The number of images for stage 1 are low compared to rest of

the stages (Table 1). Fewer images imply that the model has less instances to learn the features of these images which may be a major reason of non-detection of first stage.

However, there must be other reasons for non-detection for the third stage because the number of images seem sufficient, but performance of the model is poor. This may be because the features in the second and third stage are very similar. There are studs and gypsum panel visible in both stages. Visually the electrical wires are very slender and take very little space on the image (example: figure 7). Thus, the model predicts it to be stage 2 instead of stage 3. The work attempted to classify all major stages in dry wall construction. In contrast to Ekenayake et al. (2022), we got good results for stage 2: gypsum paneling, stage 5: both side paneled and stage 6: plastering. This distinction of stages has not been done in previous works; most researchers before having divided the drywall progress into fewer stages.

Also, classification and localization by YOLOv7 is more accurate than YOLOv4 which has more localization errors (John and Meva, 2023). The limitation of our work is that, currently we are not able to detect the framing stage accurately which has been well detected by Hamledari et. al. (2017) and Ekenayake et al. (2022). This needs further investigation for improvement of the detection process. The model which was produced by training on a small set of drywall images showed positive results for as-built stage detection of drywall. However, some challenges remain for using neural networks for object detection. The number of images should be adequate so that model can learn features for accurate prediction. If the features are similar in several stages, there is need of data augmentation techniques to be employed. To distinguish between similar stages, semantic information could also be provided to the neural network to enhance the prediction capabilities. This highlights the need for an extensive database which could store images in a semantic way. In future research, we will extend this research to include more images of the stages and employ data augmentation techniques for a robust model. Ultimately, a generalizable solution for many indoor progress detection tasks is required to enable automatic data analysis for digital twins in construction.

Conclusion

Automated construction progress monitoring is important not only from a perspective of creating real time digital situational awareness, but it forms an important part of digital twins in construction phase. In this work we identified various challenges which appear while applying modern techniques like deep learning using sensors for automatic progress monitoring. Challenges like lack of annotated dataset and lack of a comprehensive framework related to indoor construction were solved to an extent by transfer learning. Yet challenges of occlusion and semantic interpretation remain. In future research, this process should be integrated with automatic data collection from

worksite. It can also be explored how computer vision results can be augmented by combining other data sources.



Figure 5: Prediction for Visible side Gypsum Paneling



Figure 6: Prediction for Gypsum plastering



Figure 7: Prediction for Electrical wiring stage

In future works, we will explore how data from different sensors like location, temperature, humidity, and videos/images can be fused together for workflow monitoring. This is envisaged to be done by using a real time data collection and identification system involving Spot robot and algorithm developed for the use case. The Spot robot will collect videos and point cloud data of the site and these along with other sensor data will be

processed using AI. This data would be transferred to a server where processing will be done, and the processed information would be returned to a system which can help visualize the situation on the worksite. This system of visual management could be a digital board or a tablet or an app made for visual management and accessible by all stakeholders.

References

Dimitrov, A., & Golparvar-Fard, M. (2015). Segmentation of building point cloud models including detailed architectural/structural features and MEP systems. *Automation in Construction*, 51, 32-45.

Fathi, H., Dai, F., & Lourakis, M. (2015). Automated as-built 3D reconstruction of civil infrastructure using computer vision: Achievements, opportunities, and challenges. *Advanced Engineering Informatics*, 29(2), 149-161.

Hamledari, H., McCabe, B., & Davari, S. (2017). Automated computer vision-based detection of components of under-construction indoor partitions. *Automation in Construction*, 74, 78-94.

Han, K. K., Cline, D., & Golparvar-Fard, M. (2015). Formalized knowledge of construction sequencing for visual monitoring of work-in-progress via incomplete point clouds and low-LoD 4D BIMs. *Advanced Engineering Informatics*, 29(4), 889-901.

Han, K., Degol, J., & Golparvar-Fard, M. (2018). Geometry-and appearance-based reasoning of construction progress monitoring. *Journal of Construction Engineering and Management*, 144(2), 04017110.

John, A., & Meva, D. (2023). Comparative Study of Various Algorithms for Vehicle Detection and Counting in Traffic. In *Advancements in Smart Computing and Information Security: First International Conference, ASCIS 2022, Rajkot, India, November 24–26, 2022, Revised Selected Papers, Part II* (pp. 271-286). Cham: Springer Nature Switzerland.

Kenley, R., & Seppänen, O. (2010). *Location-Based Management for Construction: Planning, Scheduling and Control*. Spon Press.

Krizhevsky, Alex (2009). Learning Multiple Layers of Features from Tiny Images.

Masood, M. K., Aikala, A., Seppänen, O., & Singh, V. (2020). Multi-Building Extraction and Alignment for As-Built Point Clouds: A Case Study With Crane Cameras. *Frontiers in Built Environment*, 6, 581295.

Mirzaei, K., Arashpour, M., Asadi, E., Masoumi, H., Bai, Y., & Behnood, A. (2022). 3D point cloud data processing with machine learning for construction and infrastructure applications: A comprehensive review. *Advanced Engineering Informatics*, 51, 101501.

Pan, S. J., & Yang, Q. (2010). A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10), 1345-1359.

Paneru, S., & Jeelani, I. (2021). Computer vision applications in construction: Current state, opportunities & challenges. *Automation in Construction*, 132, 103940.

Pregnolato, M., Gunner, S., Voyagaki, E., De Risi, R., Carhart, N., Gavriel, G., ... & Taylor, C. (2022). Towards Civil Engineering 4.0: Concept, workflow and application of Digital Twins for existing infrastructure. *Automation in Construction*, 141, 104421.

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 779-788).

Roh, S., Aziz, Z., & Pena-Mora, F. (2011). An object-based 3D walk-through model for interior construction progress monitoring. *Automation in Construction*, 20(1), 66-75.

Sacks, R., Brilakis, I., Pikas, E., Xie, H. S., & Girolami, M. (2020). Construction with digital twin information systems. *Data-Centric Engineering*, 1, e14.

Seong, H., Choi, H., Cho, H., Lee, S., Son, H., & Kim, C. (2017). Vision-based safety vest detection in a construction scene. In *ISARC. Proceedings of the International Symposium on Automation and Robotics in Construction (Vol. 34)*. IAARC Publications.

Seppänen, O., & Görsch, C. (2022, July). Decreasing waste in Mechanical, Electrical and Plumbing work. In *Annual Conference of the International Group for Lean Construction* (pp. 84-94). IGCLC.net.

Son, H., & Kim, C. (2010). 3D structural component recognition and modeling method using color and 3D data for construction progress monitoring. *Automation in Construction*, 19(7), 844-854.

Wang, C. Y., Bochkovskiy, A., & Liao, H. Y. M. (2022). YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. *arXiv preprint arXiv:2207.02696*.

Xin, S., Nousias, S., Kutulakos, K. N., Sankaranarayanan, A. C., Narasimhan, S. G., & Gkioulekas, I. (2019). A theory of Fermat paths for non-line-of-sight shape reconstruction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 6800-6809).

Yang, J., Park, M. W., Vela, P. A., & Golparvar-Fard, M. (2015). Construction performance monitoring via still images, time-lapse photos, and video streams: Now,

tomorrow, and the future. *Advanced Engineering Informatics*, 29(2), 211-224.

Zhao, J., Pikas, E., Seppänen, O., & Peltokorpi, A. (2021). Using real-time indoor resource positioning to track the progress of tasks in construction sites. *Frontiers in Built Environment*, 7, 661166.