Towards Construction Progress Estimation Based on Images Captured on Site

Peter Hevesi[1(✉)], Ramprasad Chinnaswamy Devaraj[2], Matthias Tschöpe[1], Oliver Petter[1], Janis Nikolaus Elfert[1], Vitor Fortes Rey[1], Marco Hirsch[1], and Paul Lukowicz[1,2]

[1] German Research Center for Artificial Intelligence (DFKI), Kaiserslautern, Germany
{peter.hevesi,matthias.tschoepe,oliver.petter,janis_nikolaus.elfert, vitor.fortes_rey,marco.hirsch,paul.lukowicz}@dfki.de
[2] University of Kaiserslautern, Kaiserslautern, Germany
ramprasad.devaraj@gmx.de, lukowicz@cs.uni-kl.de

Abstract. State of the art internet of things (IoT) and mobile monitoring systems promise to help gathering real time progress information from construction sites. However, on remote sites the adaptation of those technologies is frequently difficult due to a lack of infrastructure and often harsh and dynamic environments. On the other hand, visual inspection by experts usually allows a quick assessment of a project's state. In some fields, drones are already commonly used to capture aerial footage for the purpose of state estimation by domain experts.

We propose a two-stage model for progress estimation leveraging images taken at the site. Stage 1 is dedicated to extract possible visual cues, like vehicles and resources. Stage 2 is trained to map the visual cues to specific project states. Compared to an end-to-end learning task, we intend to have an interpretable representation after the first stage (e.g. what objects are present, or later what are their relationships (spatial/semantic)). We evaluated possible methods for the pipeline in two use-case scenarios - (1) road and (2) wind turbine construction.

We evaluated methods like YOLOv3-SPP for object detection, and compared various methods for image segmentation, like Encoder-Decoder, DeepLab V3, etc. For the progress state estimation a simple decision tree classifier was used in both scenarios. Finally, we tested progress estimation by a sentence classification network based on provided free-text image descriptions.

Keywords: Construction progress estimation · Neural networks · Computer vision

This work has been partly funded by the Federal Ministry of Education and Research of Germany (BMBF) within the framework of the project ConWearDi (project number 02K16C034).

L. Peñalver and L. Parra (Eds.): Industrial IoT 2020, LNICST 365, pp. 141–161, 2021.
https://doi.org/10.1007/978-3-030-71061-3_9

1 Introduction

Connecting digital models and plans with physical reality has been a key application of Internet of Things. In the industrial domain it is at the core of the Industry 4.0 concept that is currently driving a wave of transformation that is often compared to the original industrial revolution [6]. Concrete applications range from monitoring progress of specific tasks to optimally support the worker and prevent errors, through process optimization to predictive maintenance.

While IoT has been very successful in the industrial domain, it has so far had much less impact on the construction sector. This is certainly not due to the lack of potential applications. Construction planning and reporting is increasingly digital. Connecting such digital plans and reports to the physical reality on the construction site is today done mostly manually which is not only time consuming but also error prone. Process monitoring is another important and necessary step for management of large construction projects and their resource optimization. Again, classical analogue "pen and paper" approaches are time consuming and error prone.

The problem with using IoT to connect construction sites to the digital domain is a combination of high instrumentation effort with only limited structure and high degree of dynamism. Thus a factory floor is in most cases a very well defined, highly structured environment with complex technical infrastructure into which sensors can be easily integrated. The infrastructure and structure tends to remain unchanged for long periods of time. As a consequence many processes taking place on the factory floor are highly structured. A construction site on the other hand is set up temporally with limited infrastructure. Sites tend to significantly differ from each other and evolve. As a consequence work processes tend to be much more adaptive.

As a consequence of the instrumentation difficulties the use of image processing techniques, in particular in conjunction with aerial, drone-based surveillance of construction sites is a growing field. The potential market value of business services that may benefit from drones has been predicted to reach several billion dollars [14]. Drones have been used in physical infrastructure (energy, roads, railways, oil and gas, construction) to examine terrain, record progress, inventory assets, inspect facilities for maintenance [14]. Drone surveillance can be eventually automatized (in some countries, there are legal restrictions), allowing a cost effective way to capture the project state frequently.

In previous work, we explored site monitoring for the earth-movement phase of projects using various sensors in the participating vehicles [3]. Building on the experience from that previous work in this paper we investigate how the increased accuracy of image based object detection techniques can be leveraged to facilitate processes tracking much like embedded sensors allow process tracking on factory floors. Specifically we aim to infer the state of a building process as defined by a human expert from images of the construction site.

The idea is based on the observation that domain experts are often able to quickly assess the current state of the site by visual inspection. This implies the existence of visual cues correlating strongly with specific steps in the process [10].

We propose a two-stage model leveraging state-of-the-art components representing human experts knowledge about project states in different types of construction processes. The first stage involves image processing to extract information about the visual cues related to the targeted process. As we present, this step can be implemented by different types of methods, e.g., semantic segmentation or object detection models. The second stage focuses on predicting the current project state based on the detected object and resources on the images. For this work, we selected two scenarios to evaluate and present the approach:

1. Road Construction
2. Wind Turbine Construction

While this model alone is insufficient to replace complex monitoring systems involving real time sensor data collection, 3D scanning and comparison with Building Information Models (BIM), it is a lightweight approach that aims to map an experts' capability to judge project states based on a single 2D image taken from the right perspective.

2 Related Work

2.1 Use Case

Sensor-based monitoring of road constructions has been proposed in the literature [12,13], where the authors use RFID and GPS information to estimate the progress. In other approaches, activity recognition on movement and location data of vehicles was used to detect progress in the earth movement operations [3].

The authors of [1] propose an advanced recording setup using autonomous drones to collect real time 3D information and compare these with the available building information model. In contrast to our approach, this requires the collection of very accurate data points across the construction site and assumes the availability of accurate planing models.

Unmanned Aerial Vehicles (UAV) have been used by [5] for calculating stockpile volume and pavement characteristics. In another paper, the authors use convolutional neural networks (CNNs) to detect existing roads from UAV images [8]. Similar techniques are proposed by [24] to use UAVs to automatically detect and assess the health condition of civil infrastructure such as bridges and pavements. The authors of [22], used data from UAVs to progress estimation based on height profile in road construction scenarios.

2.2 Methods

In our approach, we build upon state-of-the-art models for image segmentation, object detection and sentence classification. For the image segmentation task, we included networks like AdapNet [21], PSPNet [27], GCN [16], SegNet [2] and DeepLabV3 [4] in our work. We used the YOLOv3-SPP implementation by Jocher et al. [7], which is an improvement of the original YOLO, which was

invited by Redmon and Farhadi [17–19]. For these networks, there are various pre-trained model weights available, that can be leveraged when training on a data set with low amount of examples and were therefore well suited for our concept.

A combination of a deep vision CNN with a language generating RNN was proposed by Vinyals et al. In their "Show and Tell" paper [23], the authors demonstrated that their NIC (Natural Image Caption) approach significantly outperformed previous state-of-the-art solutions in automatic image captioning. Consecutive research in this field has led to steady improvements in captioning quality [25,26]. Hence, we explored an initial approach towards using natural language descriptions as input for the progress estimation. It should be noted that the majority of the methods described in literature aim at captioning a wide variety of images from different contexts as opposed to the use cases in this work with a low diversity of meaningful scenarios.

3 Scenarios

Fig. 1. Example scenarios: road construction (left) and wind turbine assembly (right). Most process steps involve a different set of vehicles, machines, materials or parts that can be observed on the image. A person familiar with the process often can identify the currently performed process step by observing only a single image. [1]sample images from unsplash.com, [2]sample images from pixabay.com

The approach could be applied to every real world scenario to estimate progress where there are existing visual cues that are specific to a given progress step. Many construction processes meet that criterion and are therefore ideal candidates for an initial evaluation. Most of them are of a linear nature, as previous steps must be completed before the next ones start. Furthermore, the progress of construction as well as material, vehicles, tools and workers, that are involved in the construction process, can be visually observed in every state. This is especially true, because just in time delivery and deployment is highly relevant in

construction projects, due to the costs and expenditure that are associated with storing materials at the construction site and renting vehicles.

In this work, we considered two different construction processes to prove the feasibility of our approach; the construction process of roads and the installation process of wind turbines. Some example images found on the internet including different states in these two scenarios are depicted on Fig. 1.

3.1 Road Construction

The first scenario is the process of road constructions, more specifically the construction of roads with flexible pavement asphalt. The construction process of roads differs, based on the pavement type that is used. The focus on asphalt pavement roads was selected because of the sufficient complexity and visual variations during the process steps. A typical process for a road construction is comprised of five sequential stages, the first five in the list provided below. In order to cover the scenario of road reconstruction and maintenance, we have included milling process as part of the construction stages. Milling is used to remove the damaged layers of the road and enables thorough repair by removing surface irregularities. Depending on the depth of surface removed during reconstruction, the other stages are carried out sequentially to complete the road construction.

1. Preparation - Excavation
2. Subgrade layer construction
3. Subbase layer construction
4. Base layer construction
5. Pavement layer construction
6. Milling

The assumption here is, that each of these steps can be usually identified by the occurrence of specific objects found on the construction site during its execution. We divided these objects into three main categories. These categories are 1) construction workers 2) vehicles or machines used during the process and 3) construction resources and materials. The category *construction worker* (or person) does not include any further sub-classes for now.

The following construction vehicles were identified to be involved in the process during individual steps of applying asphalt pavements: *Excavator, Dump truck, Bulldozer, Motor-grader, Paving machine, Distributor, Milling machine, Road roller.*

The construction materials relevant in the use-case are *Sand, Asphalt, Aggregate, Natural soil or clay and Gravel.*

The exact mapping of resources to progress states are not presented in this paper, since in the idea this representation is learnt automatically during the training step.

3.2 Wind Turbine Installation

As a second application scenario, the installation process of wind turbines was selected. Similar to the road construction, this involves a number of steps performed in sequence. The typical construction process for this scenario consists of the following stages:

1. Site preparation - Excavation
2. Foundation stage: (a) Inserting steel reinforcement, (b) Pouring concrete, (c) Curing and compacting the foundation with soil
3. Installation of Tower sections
4. Rotor Blades and Generator Installation (top): (a) installation of Nacelle section, (b) Installation of Blades to Rotor hub, (c) Installation of Rotor

During these steps, various transportation vehicles and heavy machinery are utilized. For the progress estimation, following construction and transport vehicles were identified to be relevant:

- For construction site preparation and to lay foundation: Excavator, Dump truck, Bulldozer, Road roller, Concrete mixer truck, Concrete pump truck
- For transportation: flatbed or dropdeck truck
- For installation: Heavy capacity crane (>150 ton), Medium capacity crane (≥20 ton and ≤150 ton), Low capacity crane (<20 ton)

In addition to the vehicles, the presence of different parts and materials is assumed to be also relevant to the progress estimation. For the wind turbine assembly these parts are the following: *Reinforcing steel, Concrete basement, Embedded ring, Tower parts (e.g. in horizontal, tilted or upright positions), Nacelle, Rotor hub, Blade - can also subdivided into two sub-classes based on horizontal or tilted/vertical positions.*

Similar to the road construction use case, a direct mapping of these categories to the process step is not provided, but learnt during the training phase.

4 Progress Estimation

4.1 Concept

The core idea for the progress estimation is to use a single image that contains a representative view of the site. Images, such as those on Fig. 1, provide enough information for understanding the current state in the progress towards completion.

We propose a model divided into two main stages. The first stage is responsible of extracting the available visual information from the image. In an intermediate step, the image processing results are filtered (e.g. removing false positives based on low confidence or size filters) and mapped to the format expected by the second stage. The second stage receives the previously extracted information from the image processing (e.g., what machines and resources can be seen),

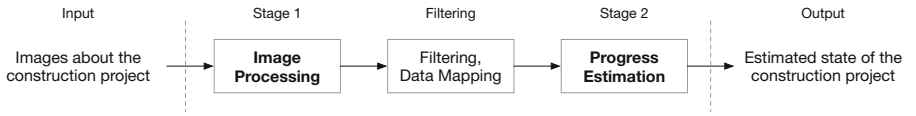

Fig. 2. Overview of the estimation pipeline. The two major parts of the system are an image processing network (e.g., for semantic segmentation) and a stage classification to predict the project's state.

and maps these to the project's state. An overview of the proposed pipeline is presented on Fig. 2.

The main reason for the division into two separate stages is to have a modular, extendable system, where the parts can be trained on different data sets. By doing so, the image processing part can be independently optimized to recognize construction resources and machines in general, and the second stage can be just trained scenario specific to evaluate presence, co-location and other spatial relations of those. Ideally, this would lead to an easier knowledge transfer to new scenarios.

4.2 Stage 1: Image Processing

Goal of the first stage is to extract relevant visual information from the provided image by detecting machines, materials and other for the scenario relevant resources. For this step to provide meaningful output, the image should fulfill minimum requirements like:

- Image quality (e.g., brightness, contrast, sharpness) should be good enough to be able to recognize relevant objects on it.
- The image should be taken from an appropriate angle and distance that captures all relevant parts of the scene.
- Vehicles and objects should not be occluded by each other in a way that prevents useful detection of other items.

For this processing step, there are different alternative approaches to extract data from images:

- Object detection: detecting specific object's bounding boxes (and with it presence) on the image.
- Semantic segmentation: assigning a class to each pixel of an image, ideally segmenting it into meaningful regions.
- Image captioning: generating a human readable textual description of the scene.

Depending on the complexity of the scenario, the appropriate method can be different. Simple object detection models are well suited to predict the presence of objects on the image. For detecting surfaces (e.g. asphalted road segment), or estimating more detailed object information (like orientation, size, relation to

other objects), image segmentation models are eventually a better choice. In this paper, we focused on evaluating methods for semantic segmentation and object detection in the scenarios. Image captioning models were not included in this initial work.

Object Detection. For object detection, we used the YOLOv3-SPP implementation by Jocher et al. [7] and modified the code for our purposes. We computed new anchors, added random crop, motion blur and random noise as new augmentation methods, and reduced the input images to a size of 864×486. We also used the pretrained weights from Jocher et al. [7], which were trained on the MS COCO data set [11].

An advantage for this method is the simple and fast annotation process, compared to semantic segmentation. A disadvantage can be the limited information about the objects. Sometimes the bounding box is very large, even if the object is rather small, e.g. if a blade occurs diagonally in the image.

Image Segmentation. In this task, the goal is to identify regions of the image belonging to different entities. Typically the output is a class label for each image pixel. Neighbouring pixels with the same class can be merged into a bigger region. For image segmentation, we compared network architectures like *PSPNeT*, *DeepLabV3*, *AdapNet*, *Global Convolutional Network (GCN)* and *Encoder-Decoder* network. The models were pretrained using ILSVRC2012 data set [20].

When working well, the output of the model is able to provide accurate object boundaries, and thus for example could be used for estimating the object's pose. On the other hand, image segmentation methods are typically more resource intensive in training and inference phase as well. Annotation task is typically a highly time consuming task, since requires accurate ground truth information for the whole image.

Some expected output examples for the semantic segmentation are shown on Fig. 3.

Filtering. In this intermediate step, the outputs of the first stage are filtered and prepared as an appropriate input for the second stage. In particular, one step here is an attempt to remove some of the false positives. In case of object detection network, candidates can be filtered by confidence of the network for the outputted label. Removing the objects with low predicted confidence, typically reduces the amount of wrong detections. For image segmentation, we applied simple size filters, where certain objects are known to have at least a specific size, e.g., an excavator need to contain more than just a couple of pixels to be accepted as an excavator instance.

To input the intermediate results into a simple classification model, we generated feature vectors for each image in a format such as $x = [0, 1, ...0, 1]$, where each position in the array represents a specific entity that could be relevant for the scenario (e.g. *crane*, etc.). At the given index of the feature vector, we use

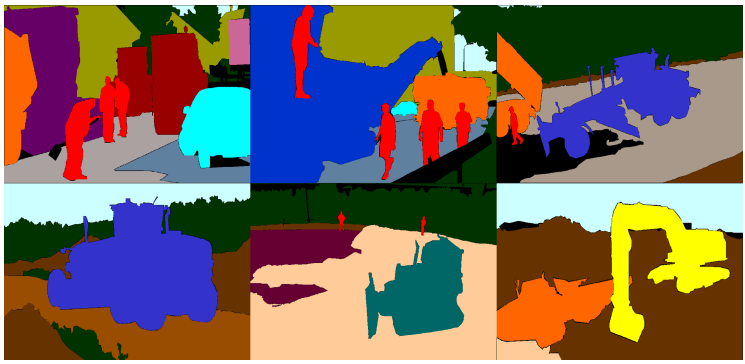

Fig. 3. Examples for semantically segmented images in the road construction scenario. Different colors represent parts of individual classes (e.g. excavator, worker, asphalt etc.).

the value 1 if the object is present on the image and 0 if not. For both results of the segmentation, and object detection, we used this simple "presence" representation in this work.

4.3 Stage 2: Progress Estimation

In the second stage, the progress of the construction process is determined using the information from images in the previous stage. An additional label is introduced in addition to the project's progress classes called "Need Additional Information" (NAI). The label is used in case the provided input for stage 2 is insufficient to assign one of the project's states to it, for example when the image was taken from an unfavorable perspective where the image detection fails to predict one of the key items.

Progress Estimation Using a Decision Tree. To predict the class label of the project stage, we trained a model using the binary feature vector provided by the filter step of stage 1 as an input. In preliminary experiments, we tried various different common classification algorithms and could not observe any significant difference in performance. The results shown in this paper are from a simple decision tree classifier we chose for its intuitive representation of different object presence to project stage label.

Progress Estimation with a Sentence Classification Network. While stage 1 presented in this paper solely provides a binary feature vector, we tested an initial approach using natural language descriptions as an intermediate representation. Descriptions of the images created by humans are used as an input for an alternative stage 2 model consisting of a sentiment analysis for sentences

utilising convolutional neural networks, which have shown to be successful for these type of tasks in the past [9].

Encoding the image information into a free text adds information about the relationship of objects to each other which can be interpreted by a suitable model. Additionally, the form of natural language enables an alternative approach in case camera usage is not possible or feasible. Anyone on the field can provide a short description of the scene and the expert knowledge about the process is added by the stage 2 model.

5 Evaluation

5.1 Object Detection for Wind Turbine Installation

Performance of a custom trained YOLOv3-SPP network was evaluated on detecting relevant entities on images taken on wind turbine constructions.

Data Set. For this evaluation, we created a custom data set, acquired from videos available on the internet. To capture the most important categories, as described in Sect. 3, we annotated the following seven classes with bounding boxes on the image:

crane := {Low capacity crane, Medium capacity crane, Heavy capacity crane}
tower := {Tower-horizontal, Tower-tilted, Tower-vertical, Embedded ring}
blade := {Blade-horizontal, Blade-tilted}
motor := {Nacelle, Rotor hub}
fbm := {Excavator, Dump truck, Bulldozer, Road roller, Concrete mixer truck, Concrete pump truck}
trans := {Flatbed truck, Dropdeck truck}
found := {Concrete Foundation, Reinforcing steel, Concrete}

where fbm is our abbreviation for *foundation building machine*, trans stands for *transport truck* and found for *foundation*.

In total, we labeled 911 images, containing 2285 objects from 31 videos. These objects are allocated to the above classes as follows: crane 577, tower 567, blade 337, motor 315, fbm 242, trans 114 and found 133. We used a random train-val-split with 90% training and 10% validation images. Our test set consists of frames of four independent videos, i.e. there is not a single frame in the test set that is also in the training set or validation set. Under those preconditions, we picked the test-videos in such a way, that the class-ratio distribution of the test-set is similar to the class-ratio distribution of the whole data set. This leads to a test-set with 101 images containing 72 crane, 68 tower, 38 blade, 28 motor, 23 fbm, 17 trans and 17 found objects. Notice, the column *support*, used in Table 1 is described in [15] and denotes the number of positive ground truth labels, which is equivalent to TP+FN. However, since we prefiltered some predictions by setting the Class-Specific Confidence Score to 0.1, not all of these numbers match to the numbers mentioned above.

Fig. 4. Examples of detected objects by YOLOv3-SPP after re-training with our annotated images from wind turbine construction videos. Instances of the class "foundation building machines" (e.g. excavator) are as well detected on road construction images.

Results. In the current phase of evaluation, we reduced the object detection output to the presence of an object on the image. This leads to a multi-label classification problem where we can use Precision, Recall and F_1 score for each class. The results for this are shown in Table 1. Best results were achieved for the *crane* class with a Precision, Recall and F_1 score of over 90%. Since *cranes* were the most commonly represented and visually distinct class in the data set, this result was expected.

The second last row (weighted mean) in Table 1 indicates the weighted metrics, i.e. taking the respective sums of TP, FP, FN and TN values from the seven classes and compute the metrics with those totals. Thus, the metrics of smaller classes (like transport truck) have smaller TP, FP, FN and TN values and therefore less impact of the metrics. The last row (mean) averages the metrics over all seven classes, and therefore weights each class equally. In addition to Table 1 Fig. 4 shows some object detection results applied on each single image from Fig. 1. Since we trained YOLOv3-SPP on a 16:9 ratio, we embedded each image from Fig. 1 to a 16:9 background. This was especially important for the upright image.

Considering that our data set is rather unbalanced and relatively small, the results indicate that with proper training set sizes, we can achieve good results detecting construction process related objects.

5.2 Semantic Segmentation for Road Construction

Semantic segmentation methods for detecting construction vehicles and materials were evaluated on a custom image data set for road constructions.

Data Set. To evaluate the performance of the networks as given in Sect. 4.2, a custom data set on road construction resources is created by acquiring image frames from videos available on the internet. The construction vehicles and materials as described in Sect. 3.1 are of unique nature and are important in

Table 1. Step1: evaluation results, object detection

	Precision	Recall	F_1 score	Support
Crane	0.98	0.93	0.95	70
Tower	0.78	0.77	0.77	66
Blade	0.70	0.70	0.70	37
Motor	0.46	0.59	0.52	27
Fbm	0.75	0.82	0.78	22
Trans	0.29	0.24	0.26	17
Found	0.92	0.65	0.76	17
Weighted mean	0.75	0.75	0.75	256
Mean	0.70	0.68	0.69	256

determining the current ongoing process at construction site. Therefore are annotated as separate class labels. Further, the background information observed during road construction is categorized into one of the following class labels: 1) Sky, 2) Vegetation, 3) Signboard, 4) Car, 5) Barricade 6) Building. The parts of the image that does not belong to the above classes are annotated as Unlabeled.

A total of 330 images of size 800×576 covering all processes involved in road construction are pixel-wise annotated. Data augmentation methods are followed to improve the data set size. Augmentation techniques such as varying brightness intensity, flipping images horizontally and introducing blur and noise in the images are used. With this, the size of the data set is increased to a total of 1650 images. Out of which 1200 images are used for training and 300 images for validating. The test set consists of 150 images with an even distribution of 25 images per road construction stage as described in Sect. 3.1. The image frames in test set are not used for training or validation.

Results. Performance of a semantic segmentation model is normally evaluated using metrics such as Intersection of Union(IoU), Precision, Recall and F_1 score. Due to class imbalance issue in the data set, pixel accuracy is not used to evaluate the model. As the road construction scenario involves determining several classes as described in Sects. 3.1 and 5.2, it is a multi-class segmentation problem. Therefore, Mean Intersection of Union (mIOU) of the image is calculated by taking the IoU of each class and averaging them.

The performance of the networks on the road construction data set are provided in Table 2. The mIoU values provided in the second column from the left suggests that Encoder-Decoder network performs better than the other networks on the custom road construction data set. Since the Stage2: Progress estimation relies on the accuracy of predicted construction resources from this stage. It is important to obtain good predictions. Considering the size of the data set, the results are significant and can be further improved by increasing the training set size and covering more possibilities of real world scenarios.

Table 2. Image segmentation results using different network types on the road construction data set

	mIOU	Precision	Recall	F_1 score
AdapNet	0.65	0.88	0.87	0.87
PSPNet	0.64	0.84	0.85	0.84
Encod.-Decod	0.75	0.94	0.93	0.93
GCN	0.49	0.70	0.72	0.69
DeepLabV3	0.33	0.57	0.56	0.53

5.3 Progress Prediction - Road Construction

For evaluating the second stage in the road construction scenario, we defined two test cases, when generating the feature vectors:

1. using perfect image processing results (the ground truth labels for the images)
2. using predicted objects on the images by the best performing image segmentation network (Encoder-Decoder)

Data Set. For evaluating the second stage, we did not use any of the augmented images. In total we had 330 images with available progress labels. These labels correspond to the 6 possible progress states of road constructions as listed in Sect. 3.1. For every target class, we selected half of the available images randomly to be in the training set, and the rest was dedicated to be used for training. In total 165 images were used in training the classifier.

For testing the performance of the classification using the results of the image processing stage, we only had 30 images that were in the test set of the image processing and are original images without data augmentation. These 30 images include 5 examples for every project steps, which led to a rather small data set. Therefore, we let the training of the second stage run with the ground truth detections and used the 30 detections solely for testing.

Results. The trained progress classification decision tree model could provide a perfect classification when tested with feature vectors generated out of the ground truth objects (filtered and prepared image segments). These results are displayed in Table 3. This result for road construction was not unexpected, since we assumed an perfect mapping of resources and machines to the target classes beforehand.

The results for applying the progress estimation on realistic output of the image processing stage are listed in Table 4. Here, we used the results of the *encoder-decoder* network, since this seemed to be the most promising image segmentation method for the use case. The progress classification has only one mistake, a confusion between excavation and sub-base stages, which can be rooted

Table 3. Progress prediction results for road construction assuming perfect object detection results

	Precision	Recall	F_1 score	Support
Base	1.00	1.00	1.00	17
Excavation	1.00	1.00	1.00	30
Milling	1.00	1.00	1.00	29
Paving	1.00	1.00	1.00	31
Subbase	1.00	1.00	1.00	28
Subgrade	1.00	1.00	1.00	30

back to a partly wrong image segmentation result. Other mistakes from the segmentation did not influence the final performance. The small data set however only allows limited significance and will be subject of future work to further test with more diverse scenes.

Table 4. Progress prediction results for road construction using predictions from the Encoder-Decoder network's output as a feature vector

	Precision	Recall	F_1 score	Support
Base	1.00	1.00	1.00	5
Excavation	0.83	1.00	0.91	5
Milling	1.00	1.00	1.00	5
Paving	1.00	1.00	1.00	5
Subbase	1.00	0.80	0.89	5
Subgrade	1.00	1.00	1.00	5

5.4 Progress Prediction - Wind Turbine Construction

For evaluating the second stage in the wind turbine construction scenario, we defined two test cases, when generating the feature vectors:

1. Assuming perfect image processing results by using the ground truth labels
2. Using predicted objects by the fine-tuned YOLOv3-SPP network

Data Set. For each of the 911 images in the wind turbine data set, we assigned the process step label as one of the following classes:

- Excavation: earthwork phase of site preparation
- Foundation: building the foundation
- Tower Installation: building the tower parts

– Rotor Installation: assembly of the rotor blades, mounting of nacelle and rotor to the top of the tower
– NAI: the image does not contain enough information to be able to assign it to a single process step

For test case 1, we performed a split of the data set randomly selecting 319 instances for test to contain approximately 35% of each class and the remaining instances for training. The feature vector for each instance was created using the list of annotated objects on the image - to see how well the classifier performs with no errors coming from the first stage.

For test case 2, we wanted to evaluate the classifier under realistic conditions, where we do not have image annotations at all. For this test, we used object detection results of the YOLOv3-SPP on the test set for images (not included in the training of the network's weights). This was the basis for training and testing the second stage completely. Out of the 256 instances (feature vector representing the detected object on the image and corresponding progress class label), we selected randomly 104 for the test-set, and trained a new decision tree classifier on the remaining 152 examples.

Table 5. Progress prediction results for wind turbine construction assuming perfect object detection results used as an input

	Precision	Recall	F_1 score	Support
Excavation	0.71	0.97	0.82	30
Foundation	0.84	0.77	0.80	60
Tower installation	0.93	0.54	0.68	98
Rotor installation	0.57	0.92	0.70	88
NAI	0.67	0.37	0.48	43

Results. Results for test case 1 are shown in Table 5. Excavation and Foundation classes achieve an F_1 score of approximately 0.8, Tower and Rotor installation classes are around 0.7. A look at the confusion matrix on Fig. 5 reveals that these two pairs of sequential process step's classes are also responsible for most of the *confusions*. Probable reason for these inaccuracies, is that the simplified information about presence of objects is not enough anymore, to perfectly distinguish the classes and especially in the early project phases, similar vehicles (like excavator) are still in use or at site, when next steps are performed.

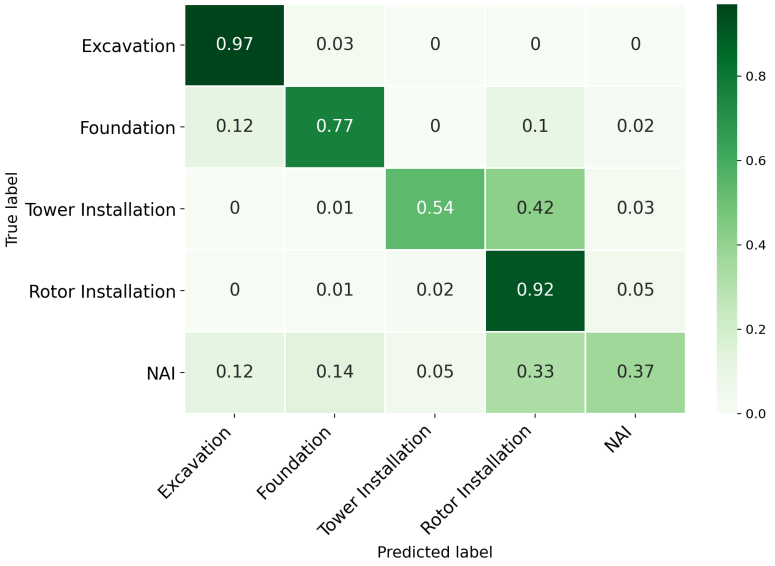

Fig. 5. Confusion matrix of the progress predictions for the wind turbine data set assuming perfect object detection results.

Table 6. Progress prediction results for wind turbine construction using detected objects by a custom trained YOLOv3-SPP network

	Precision	Recall	F_1 score	Support
Excavation	0.67	1.00	0.80	20
Foundation	0.92	0.67	0.77	18
Tower installation	0.65	0.76	0.70	29
Rotor installation	0.61	0.55	0.58	20
NAI	0.44	0.24	0.31	17

In test case 2, using only the predicted object detections for feature vector generation, we performed the same training workflow. Results are summarized in Table 6 and in the confusion matrix on Fig. 6. With exception of the class *Rotor Installation*, that has worse results, the scores are in a similar range as in test case 1.

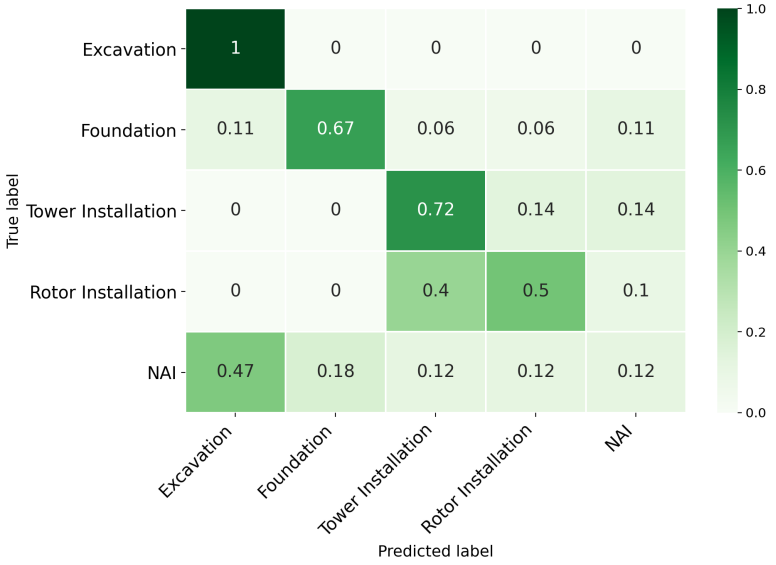

Fig. 6. Confusion matrix of the progress predictions for the wind turbine data set when using objects detected by YOLOv3 model.

5.5 Wind Turbine Construction - Using Image Captions

Data Set. We selected a subset of images throughout all the stages from 4 different construction sites and let 3 people describe the image content in text form leading to 633 annotations for 211 images, like the following examples.

- "An excavator digs a hole into a field."
- "A construction sign, with a tower section hanging on a heavy capacity crane next to the vertical tower parts in the background."
- "A road roller is behind an excavator, that is performing earth moving operations."
- "A blue heavy capacity crane is next to a half-built tower."

It should be noted that the annotators had no expert knowledge about the processes nor were they associated with the construction industry but they have been shortly briefed for the task. In addition to explaining the purpose of the labels, they were briefly explained what the construction scenario is about. The description should be focused on the visible objective representation of the image, with no interpretation or mentions regarding the process state. Additionally, they were asked to use predefined names for the key objects, like "heavy capacity crane" or "rotor blade" whenever possible. The total description text length was limited to a maximum of 200 characters per image, which seemed to be more than sufficient for most of the cases. Apart from that, the annotators could write freely without any restrictions regarding language complexity sentences or grammar.

Results. Table 7 shows the averaged results of using a sentence classification CNN, trained with the image captions from two people and tested on the descriptions of the remaining person. The respective confusion matrix can be seen in Fig. 7. Generally, the network performs well with an 0.84 F_1 score averaged over the 4 project classes. Interestingly, the NAI class assigned in the "expert assessment phase" is more often assigned to one of the other classes when using non-expert descriptions compared to the object detection results. Hence, the network could not properly distinguish if additional infos are needed. Apart from the NAI class, confusion almost only happen in subsequent process steps.

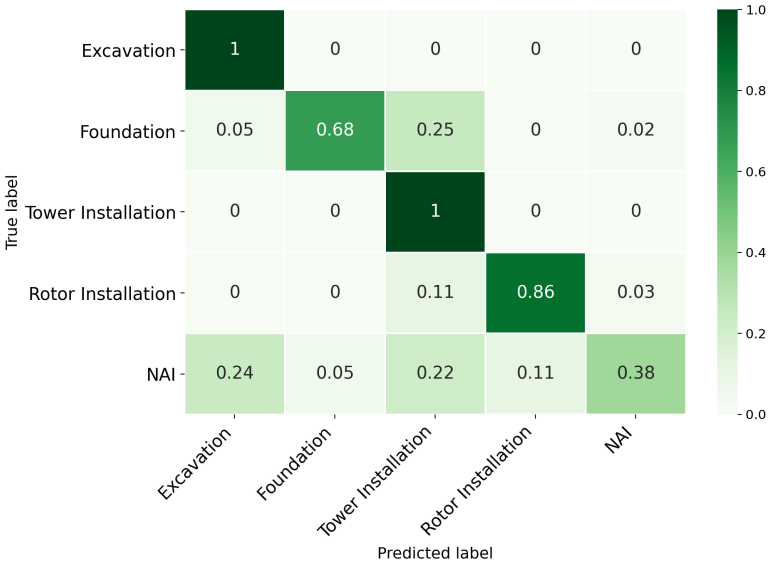

Fig. 7. Confusion matrix of the progress predictions with sentence classification CNN using image captions.

Table 7. Progress prediction results for wind turbine construction using a sentence classification CNN with image captions

	Precision	Recall	F_1 score	Support
Excavation	0.77	1.00	0.87	36
Foundation	0.93	0.68	0.78	40
Tower installation	0.74	1.00	0.85	62
Rotor installation	0.89	0.86	0.87	36
NAI	0.88	0.38	0.53	37

6 Conclusion

Construction projects present in most cases a changing and for IoT systems an often challenging environment. This makes remote, automatic monitoring difficult task. In this paper, we propose a lightweight, alternative path for automatically predicting the construction project's state based on images taken at the construction site. This initial work describes a modular, two stage model and evaluates some state-of-the-art techniques that could be used in the different stages of the pipeline.

The core idea is to apply a suitable image processing model in the first stage, that extracts a human readable representation of the relevant visual cues from the image. For simple use case scenarios, the *visual cue* can be just the presence of specific objects on the image. The second stage can be independently optimized to map the extracted cues to the project state.

For object detection tasks, we evaluated the usage of a YOLOv3-SPP network re-trained specifically to recognize key objects for a wind turbine construction use case. Even with a small amount of images in the data set (911 in total during different phases of the construction process), results look promising, with one of the classes (crane) even performing above 0.9 precision and recall values when focusing on presence of objects.

We also performed an initial evaluation of different image segmentation networks, by annotating images from the road construction scenario and comparing them to alternative models for the task. An Encoder-Decoder network could achieve precision and recall scores above 0.9 on average over all classes. The approach of image segmentation can be particularly interesting later in use cases where more information is necessary than just the presence of certain objects.

The progress detection step in the second stage utilized a simple *decision tree* classifier for both scenarios. In the road construction scenario, this approach seems to be able to provide perfect classification results assuming a perfect prediction from the image processing stage. The wind turbine scenario proved to be more challenging. However, the classifier performs reasonably well, the tower and rotor installation classes show a lot of confusion. The issue here seems to be in the nature of the underlying data, hinting that a simple presence of objects might be insufficient information to predict this project's states.

Motivated by this outcome and the idea to keep a human readable intermediate data representation, we explored if and how well manually generated free text image captions can be mapped to the project state. For this task, a sentence classification CNN was utilized, what could learn to map given input texts to the output class label. For 3 classes, this image caption interpreter achieved a F_1 score above 0.84. This shows that a free text written by a human observer can be mapped to a more abstract process state even if the observer does not have expert knowledge about the process itself.

Initial results, included in the paper, provided helpful insights for the way towards a practically useful system. The two stage model seems to be very helpful because its modular nature makes it possible to always select state-of-the-art specialized methods for the sub-tasks. Looking at the initial wind turbine

results, we assume that many scenarios will require more detailed information like object relationships (spatial and semantic). Hence, besides training on larger image data sets with a higher variety of construction related entities (material, tools, vehicles) and exploring other use case scenarios, in future work, we want to evaluate the performance of automated image captioning models and their potential of mapping to process stages.

References

1. Anwar, N., Izhar, M.A., Najam, F.A.: Construction monitoring and reporting using drones and unmanned aerial vehicles (UAVs). In: The Tenth International Conference on Construction in the 21st Century (CITC-10) (2018)
2. Badrinarayanan, V., Kendall, A., Cipolla, R.: SegNet: a deep convolutional encoder-decoder architecture for image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **39**(12), 2481–2495 (2017)
3. Bucchiarone, A., et al.: Smart construction: remote and adaptable management of construction sites through IoT. IEEE Internet Things Mag. **2**(3), 38–45 (2019). https://doi.org/10.1109/IOTM.0001.1900044. https://ieeexplore.ieee.org/document/8950968, print ISSN: 2576-3180 Electronic ISSN: 2576-3199
4. Chen, L.C., Papandreou, G., Schroff, F., Adam, H.: Rethinking atrous convolution for semantic image segmentation (2017)
5. Congress, S.S.C., Puppala, A.J.: Novel methodology of using aerial close range photogrammetry technology for monitoring the pavement construction projects. In: International Airfield and Highway Pavements Conference 2019, pp. 121–130. American Society of Civil Engineers (2019). https://doi.org/10.1061/9780784482476.014
6. Drath, R., Horch, A.: Industrie 4.0: Hit or hype? [industry forum]. IEEE Ind. Electron. Mag. **8**(2), 56–58 (2014)
7. Jocher, G., et al.: ultralytics/yolov3: Rectangular Inference, Conv2d + Batchnorm2d Layer Fusion (2019). https://doi.org/10.5281/zenodo.2672652
8. Kestur, R., et al.: UFCN: a fully convolutional neural network for road extraction in RGB imagery acquired by remote sensing from an unmanned aerial vehicle. J. Appl. Remote Sens. **12**(1), 016020 (2018). https://doi.org/10.1117/1.JRS.12.016020
9. Kim, Y.: Convolutional neural networks for sentence classification. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1746–1751. Association for Computational Linguistics, Doha, Qatar, October 2014. https://doi.org/10.3115/v1/D14-1181, https://www.aclweb.org/anthology/D14-1181
10. Kopsida, M., Brilakis, I., Vela, P.: A review of automated construction progress monitoring and inspection methods. In: Proceedings of the 32nd CIB W78 Conference on Construction IT (2015)
11. Lin, T.-Y., et al.: Microsoft COCO: common objects in context. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8693, pp. 740–755. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10602-1_48
12. Navon, R., Shpatnitsky, Y.: Field experiments in automated monitoring of road construction. J. Constr. Eng. Manage. **131**(4), 487–493 (2005). https://doi.org/10.1061/(ASCE)0733-9364(2005)131:4(487)

13. Navon, R., Shpatnitsky, Y.: A model for automated monitoring of road construction. Constr. Manage. Econ. **23**(9), 941–951 (2005). https://doi.org/10.1080/01446190500183917
14. Otto, A., Agatz, N., Campbell, J., Golden, B., Pesch, E.: Optimization approaches for civil applications of unmanned aerial vehicles (UAVs) or aerial drones: a survey. Networks **72**(4), 411–458 (2018). https://doi.org/10.1002/net.21818
15. Pedregosa, F., et al.: Scikit-learn: machine learning in Python. J. Mach. Learn. Res. **12**, 2825–2830 (2011)
16. Peng, C., Zhang, X., Yu, G., Luo, G., Sun, J.: Large kernel matters-improve semantic segmentation by global convolutional network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4353–4361 (2017)
17. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: unified, real-time object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 779–788 (2016)
18. Redmon, J., Farhadi, A.: YOLO9000: better, faster, stronger. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7263–7271 (2017)
19. Redmon, J., Farhadi, A.: YOLOv3: an incremental improvement. arXiv preprint arXiv:1804.02767 (2018)
20. Russakovsky, O., et al.: ImageNet large scale visual recognition challenge. Int. J. Comput. Vis. **115**(3), 211–252 (2015). https://doi.org/10.1007/s11263-015-0816-y
21. Valada, A., Vertens, J., Dhall, A., Burgard, W.: Adapnet: adaptive semantic segmentation in adverse environmental conditions. In: 2017 IEEE International Conference on Robotics and Automation (ICRA), pp. 4644–4651. IEEE (2017)
22. Vick, S., Brilakis, I.: Road design layer detection in point cloud data for construction progress monitoring. J. Comput. Civ. Eng. **32**(5) (2018). https://doi.org/10.1061/(ASCE)CP.1943-5487.0000772
23. Vinyals, O., Toshev, A., Bengio, S., Erhan, D.: Show and tell: a neural image caption generator. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3156–3164 (2015)
24. Wu, W., et al.: Coupling deep learning and UAV for infrastructure condition assessment automation. In: 2018 IEEE International Smart Cities Conference (ISC2), pp. 1–7. IEEE, 16–19 September 2018. https://doi.org/10.1109/ISC2.2018.8656971
25. Xiao, X., Wang, L., Ding, K., Xiang, S., Pan, C.: Deep hierarchical encoder-decoder network for image captioning. IEEE Trans. Multimed. **21**(11), 2942–2956 (2019)
26. Yao, T., Pan, Y., Li, Y., Mei, T.: Exploring visual relationship for image captioning. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) Computer Vision – ECCV 2018. LNCS, vol. 11218, pp. 711–727. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01264-9_42
27. Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J.: Pyramid scene parsing network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2881–2890 (2017)