Kira Zschiesche* and Martin Schlüter*

**Classification and object detection with image assisted total station and machine learning**

https://doi.org/10.1515/jag-2023-0011
Received March 1, 2023; accepted April 24, 2023; published online May 9, 2023

**Abstract:** This paper deals with applications of digital imaging total stations in a geodetic context using artificial intelligence (AI). We present two different use cases. The first is to minimise manual intervention by the operator by classifying images with different backgrounds. We use a developed software to control a total station extended by an industrial camera, which is used for the in-situ calibration of the camera. We show that the AI successfully tests the captured image for its suitability for further use and under which circumstances the AI fails. The second case is the detection of different geodetic targets (reflective and non-reflective). Captured images of an imaging total station are automatically checked to see whether a supposed target is shown in the image, identify it and localise it in the image. Already implemented applications for target identification are to be supported in this way and extended by further information.

**Keywords:** classification; deep learning; image-assisted total station; machine learning; object detection.

1 Introduction

In the following we evaluate the use and applicability of captured images from an Image Assisted Total Station (IATS). The cameras built into IATSs are often used to provide a live overview on the display. The user is thus offered an additional possibility to overlook the field of view without having to actively look through the telescope. At the same time, motorised total stations can be controlled via the touch field, and the instrument can be aligned. For documentation purposes, images can be saved and manually supplemented with field notes. Measurement data can be checked in real time via the display and additional points in the images can be determined in the post-process when images are taken from different directions. Up to now, traditional image processing algorithms have been used for this purpose. The built-in CMOS sensors also support the automatic target detection for identifying the spot. The spot represents the emitted laser beam of the IATS and is used to identify the direction to the target point [1].

The measurement system Modular Digital Imaging Total Station (MoDiTa) developed at the i3mainz of the Mainz University of Applied Sciences expands an IATS modularly by means of a digital industrial camera. Applications for non-contact optical Structural Health Monitoring (SHM) of technical building structures are to be evaluated with it [2]. Images from the external ocular camera are used by means of template matching in conjunction with cross correlation to observe discrete points on the object of interest [3]. Another example of using an external camera is DAEDALUS of ETH Zurich [4, 5]. Image processing algorithms such as multi-ellipse matching are used to detect targets [6]. Another design offers the advantage of a fixed camera with an instrument such as commercial IATS. The fixed camera provides constant calibration parameters as opposed to the modular version which requires calibration after reconfiguration. An early prototype is mentioned in ref. [7] and the prototype series IATSZ2 from the manufacturer Leica is mentioned in refs. [8, 9]. In ref. [10], algorithms such as blob analysis, edge detection and feature matching are cited as particularly successful for object recognition. Ref. [11] uses images from the telescope camera and target centroid detection to align the IATS and extract the target pattern using line features.

Artificial intelligence (AI) methods are increasingly being used in a wide variety of areas and are no longer limited to the software industry. The machine learns to solve a precisely defined task through extensive training. The intelligence is based on the recognition of patterns and regularities within the data. In the field of image recognition, Convolutional Neural Networks (CNNs) have proven to be an effective tool. The great success of CNNs is due, on the one hand, to the large number of training images available, which are processed under high computational power to determine the relevant features for the task. On the other hand, CNNs allow an unlimited depth of their layered architecture, which makes it possible to learn even
very complex features and defines the term Deep Learning (DL). One Machine Learning (ML) task in this context is classification. Based on matching features, the classification is made into a certain class. A distinction can be made between binary, multi-class, multi-label and imbalanced classifications. Object detection is the recognition of instances of semantic objects of a certain class. Bounding boxes are formed around the classified object. Two tasks are solved. Within the image the class is recognised and localised [12, 13].

We describe two different use cases below. The first is to minimise manual intervention by the operator by classifying images with different backgrounds. The second application is to use the cameras that have already been integrated in total stations to support or extend already proven methods for the detection of targets and to make additional information about their properties available to the user. We focus on two topics where we expect DL to provide complementary, meaningful benefits to the traditional methods currently being used successfully. We describe the two approaches with self-generated datasets and evaluate them for their practical application.

2 Classification

The following classification application refers to the Modular Digital Imaging Total Station (MoDiTa) measurement system developed at i3mainz. Here we extend a total station with an industrial camera [14, 15]. The controlling is carried out via the self-developed control software and enables both the camera and the total station to be addressed exclusively via this. The system calibration can be carried out semi-automatically directly on site. For the calibration, the crosshair for each project is determined once in the image. This is done using the traditional image processing approach that identify the longest lines in the image. The crosshair is determined with subpixel accuracy using edge detection and compensation adjustment. In order for the detection to be successful, a uniform background is required. This should not be too dark, too bright nor irregular so that the six longest lines can be determined. Interfering lines and edges must be avoided; otherwise, false detection will occur. So far, this selection has been done manually. The user has to decide for himself whether the image sufficiently represents the crosshairs. The aim is to fully automate this process. The selection of a suitable image from the current stream of images should be as automatic as possible, thus eliminating the need for a manual step. After initial detection, crosshair tracking is already highly reliable during an on-site project, which is why we do not consider ML for ongoing crosshair tracking. The complete approach ensures that all mechanical camera movements are accurately corrected computationally. This represents a binary problem. The method we use is supervised learning. The requirements for this are annotated image data, known as training data. These provide the learning goal for the machine. In the form of pairs, the learning goal is defined on the basis of examples of input and corresponding correct output. During training, the model learns to predict the correct output for new inputs [13]. Binary image classification is a comparatively simple task where the machine only has to learn whether the searched object is represented. We based the development of the image classifier on the DL library of MVTec HALCON. The software was made available free of charge as part of the “MVTec on Campus” programme (www.mvtec.com). The pre-trained model provided is trained on internal, industry-related images. Re-training thus requires a significantly smaller amount of training datasets. The pre-trained network contains several hidden layers. This makes it suitable for more complex classification tasks, but it is more computationally and memory intensive. This leads to small batch sizes. The exact structure of the network is not described [16].

2.1 Data resources and training

For the acquisition of the necessary dataset, we used the MoDiTa measuring system and its control software. We recorded the images with different industrial cameras. We took care to capture images that were as ideal as possible for the images to be classified as detectable. In these images, the complete crosshair is shown with an even and sufficiently bright background. For the dataset that is not to be used for crosshair detection, we used synthetic data and original image captures (Figure 1A–C). This dataset contains images that show the geodetic crosshairs with overlaps, overexposures, inhomogeneous backgrounds, additional lines (which can lead to false detections) or with images that are too dark for the classical digital image processing algorithm to achieve a successful result. The classification of the images into the two classes was carried out manually and checked with the support of the control software. Based on previous research [17], the original dataset is expanded. Our first results showed that the model has learnt to decide whether the image is suitable for detection based on the background. It does not check whether a crosshair is visible. Thus, a uniform light grey image is classified as suitable, even if there is no crosshair in it. To take this into account, we captured additional images that have extreme overexposure or underexposure. The images have a uniform background but represent the crosshairs inadequately. The developed
software cannot detect the crosshairs in these images using the traditional image processing programs. The aim of this extension is not to make the classification exclusively dependent on a uniform background, but to specify the concrete detectability of the crosshairs in the training data in a better way, i.e. much more clearly distinguished than before. To account for the unbalanced dataset, we determined class weights according to the number of images. To synthetically enlarge the image dataset and avoid divergence effects such as overfitting, we enlarged the dataset by 50 % using data augmentation [18]. We mirrored the images around the two axes and performed rotations up to 50 gon. The dataset was split 70:15:15 (training:validation:test). While the validation and test data are used, among other things, for performance evaluation, the largest dataset, the training images, are used to optimise the CNN. During training each image is abstracted through a series of filters as well as activation functions, summarised as weights, and downsampling strategies. The resulting feature maps assign importance and thus distinctiveness to different regions or objects in the image in the context of the global query. In supervised learning, the prediction of the CNN is compared to the ground-truth labels. Due to the given classification, the CNN can optimise the weights independently, which is why we also speak of end-to-end trained networks. The adjustment of the weighting is carried out automatically by a backpropagation of error and represents a gradient descent procedure that is specially optimised for multi-layer networks.

To control the learning process, there are so-called hyperparameters in ML. These differ from other parameters that are determined during training. Depending on the specific use case and the basic network topology, there are different hyperparameters. They influence the speed and quality of the learning process. The goal of hyperparameter tuning is to generate the best possible model based on the training images. Two important hyperparameters for tuning the filter weights are the learning rate and the momentum. The learning rate describes the influence of the gradient on the adjustment of the weights. Too low a learning rate results in an unnecessary number of iterations, while too high a learning rate results in divergence. In our case, we start with a higher learning rate and reduce it during training [19]. Momentum describes the influence of previous adjustments. Instead of using only the gradient of the current step to determine the search, the gradients of the past steps are also used to determine the direction. This dampens the oscillation [16]. Since we have a small batch size, six in this case, we determine the momentum with a high value.

We visualise the progress of the training by means of a time-dependent loss function. The function gives the mean error over a batch of image data. As long as the loss function steadily decreases and converges to the mean error...
2.2 Evaluation

To be able to evaluate the model, the model with the highest F1-Score is examined. The F1-Score represents a metric for assessing the quality of the model on the dataset. For this we need precision and recall (also called sensitivity) [20]. The F1-Score represents the harmonic mean between these two values. A value of 1 corresponds to a perfect model. The recall indicates which proportion of positive results were actually correct (Equation (1)). Precision is the ratio of all correctly predicted positive instances to all predicted instances. It indicates the proportion of positive identifications that were actually correct (Equation (2)) [20].

\[
\text{precision} = \frac{TP}{TP + FP} \quad (1)
\]

\[
\text{recall} = \frac{TP}{TP + FN} \quad (2)
\]

\[
F1 = \frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} = 0.974 \quad (3)
\]

True Positive (TP) means correctly identified, False Positive (FP) means incorrectly identified and False Negative (FN) means incorrectly rejected.

The high value shows that the amount of incorrectly classified cases is very low. By expanding the dataset with additional overexposed and underexposed images, the F1-Score marginally increased from 0.967 to 0.974.

We use heatmaps to evaluate whether the classified object features correspond to what is being searched for. These highlight in colour the relevant areas for classification (see Figure 1D–F). This does not correspond to segment but can be used for evaluation. The calculation is carried out using the “Gradient-weighted Class Activation Mapping” (Grad-CAM) method [22]. Figure 1D–F show the calculated heatmaps. The sections highlighted in colour show the relevant areas for classification for the CNN. In all three examples, the classification is performed correctly. In example (D) the areas in the background are highlighted in particular. Examples (E) and (F) show that the interfering lines or the irregular and partly too dark background are decisive for judging these images as unsuitable for the detection of the crosshairs by means of traditional image processing algorithms.

For further evaluation, we examined random samples. We compare the images classified by CNNs with conventional image processing methods. For this, we use the control software for MoDiTa. This software does not classify the images, but it directly triggers the recognition of the longest lines in the image in order to detect the crosshair using digital image processing algorithms. If this is not successful, an error message is shown to the user. Table 1 shows a comparison of images that are suitable for conventional detection and those that are not. Overall, the CNN trained with extended dataset has a higher matching rate with traditional image processing. With this CNN, better predictions can be made as to whether the captured image can be used to detect the crosshairs. Table 1 shows results from inference. In inference, the CNN evaluates previously unknown images [23]. Table 1 image (B) is classified as unsuitable by both CNNs, although a detection can be performed successfully. Here it can be seen that the combination of an unstable background with low exposure continues to be challenging for the trained CNNs. Whereas images (A) and (C) represent an improvement in classification given the original CNN. The bright and even backgrounds alone are no longer sufficient to trigger the crosshair detection and are consequently classified as unsuitable.

3 Object detection

The use of IATS is common practice. However, the use of the built-in camera or cameras is often limited. In many cases, the telescope and overview cameras are only used for documentation or to support control via touchpad. Applications of photogrammetric methods for automatic target acquisition of non-retroreflective objects by a total station, e.g., using SURF (speeded-up robust feature) or SIFT (scale invariant feature transform), are described in ref. [24]. In the geodetic context, varieties of different targets with different properties are in use, depending on the instrument and application used. These differ, among other properties, by a different offset, which ranges from several millimetres to centimetres. It is often necessary for the user to select or locate them manually. Likewise, in order to be able to use applications such as automatic targeting or electro-optical distance measurement (EDM), a target with retroreflective properties must be aimed sufficiently accurately. Automatic target detection requires a clearly visible spot, i.e. the energy must be set correctly. However, this depends on the distance, which is often unknown. With EDM, the field of view is smaller, which means that the prism must be aimed sufficiently accurately beforehand. This in turn requires automatic targeting [1, 25]. In both cases, the overview or telescope camera that have been installed could be used for support.

The aim is to distinguish between different targets and to roughly determine the position of the object being searched for. It is not a matter of locating the centre exactly.
Table 1: Comparison of classification by ML and traditional image processing. Shown are the results of the inference with the calculated confidences of the trained CNN and the result of the triggered detection using the control software of MoDiTa. We highlighted in colour the conformity of the results with traditional image processing.

<table>
<thead>
<tr>
<th>Image</th>
<th>Trained with original dataset</th>
<th>Trained with extended dataset</th>
<th>Traditional image processing</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>☑ = suitable</td>
<td>☐ = unsuitable</td>
<td></td>
</tr>
<tr>
<td><strong>A)</strong></td>
<td>☑ = 0.23</td>
<td>☑ = 0.02</td>
<td>☐ = 0.98</td>
</tr>
<tr>
<td></td>
<td>☑ = 0.77</td>
<td>☐ = 0.98</td>
<td></td>
</tr>
<tr>
<td><strong>B)</strong></td>
<td>☑ = 0.00</td>
<td>☑ = 0.12</td>
<td>☐ = 0.88</td>
</tr>
<tr>
<td></td>
<td>☑ = 1.00</td>
<td>☐ = 0.95</td>
<td></td>
</tr>
<tr>
<td><strong>C)</strong></td>
<td>☑ = 0.91</td>
<td>☑ = 0.05</td>
<td>☐ = 0.95</td>
</tr>
<tr>
<td></td>
<td>☑ = 0.09</td>
<td>☐ = 0.95</td>
<td></td>
</tr>
<tr>
<td><strong>D)</strong></td>
<td>☑ = 0.89</td>
<td>☑ = 0.68</td>
<td>☐ = 0.32</td>
</tr>
<tr>
<td></td>
<td>☑ = 0.11</td>
<td>☐ = 0.32</td>
<td></td>
</tr>
<tr>
<td><strong>F)</strong></td>
<td>☑ = 0.76</td>
<td>☑ = 0.07</td>
<td>☐ = 0.93</td>
</tr>
<tr>
<td></td>
<td>☑ = 0.24</td>
<td>☐ = 0.93</td>
<td></td>
</tr>
</tbody>
</table>

For this investigation we have defined four different classes of geodetic targets. The images are intended to serve as an additional source of information and to support methods that have already been implemented. Object detection is about finding the object you are looking for and assigning it to a class. If an object is found, bounding boxes are formed and its localisation is compared with the design of the ground truth. During training, the network learns to adapt these so that they match the search results as well as possible.

3.1 Data resources and training

To collect the necessary image data, we used the MoDiTa measuring system and the internal cameras of a Leica Nova TS60 [26]. The overview camera has a field of view of 19.4° and the telescope camera a field of view of 1.5°. The industrial camera used changed and so did the field of view. The image data was captured simultaneously from all three cameras. We recorded a black and white target, a 360° prism, a circular prism and a metal sphere with different backgrounds and distances from the instrument. The distances were between 8 m and 55 m. Table 2 shows original images recorded by the different cameras at the same time. It illustrates the different fields of view and their restrictions. We limited the distance to 55 m, because beyond this distance the targets are only represented by about 20 × 20 pixel in the overview camera. Due to the change in dimensions, not enough information about the targets remains in the image. The aim is to use images from the different cameras with equal balance in the input data. We expanded the dataset again by means of augmentation. Especially deeper architectures benefit from data augmentation [27]. Flipping, histogram equalisation, blurring, sharpening, morphological transformation and cut out/random erasing were used
Table 2: Examples of original images from different cameras and distances of a black and white target. The crosshairs are clearly visible in the images from the industrial camera.

<table>
<thead>
<tr>
<th>Distance</th>
<th>Industrial camera</th>
<th>Telescope camera</th>
<th>Overview camera</th>
</tr>
</thead>
<tbody>
<tr>
<td>∼13 m</td>
<td><img src="image1" alt="Industrial camera image" /></td>
<td><img src="image2" alt="Telescope camera image" /></td>
<td><img src="image3" alt="Overview camera image" /></td>
</tr>
<tr>
<td>∼40 m</td>
<td><img src="image1" alt="Industrial camera image" /></td>
<td><img src="image2" alt="Telescope camera image" /></td>
<td><img src="image3" alt="Overview camera image" /></td>
</tr>
</tbody>
</table>

[28, 29]. In total we generated about 1000 images per class. The division (training:validation:test) is again in the ratio 70:15:15. Since the supervised learning method is used, annotations are necessary. The annotations were created semi-automatically. Since the images are captured using MoDiTa and the developed software, it was possible to determine the searched target with pixel accuracy by means of template matching during the capture. By means of cross-correlation, the searched pattern is detected in the images and the coordinate information is passed on for further use.

Before training, some parameters are set. For example, based on the input data, we determine that only one target is presented per image and therefore only one determination is required. The minimum confidence level is set at 0.7. Later in the evaluation respectively inference, we reduce this value for the assessment to 0.3. The initial learning rate is set higher and reduced in succeeding epochs. The loss function is adjusted using this gradient weighting. In this case, the batch size is four images, which are processed simultaneously. Since the batch size is small, the momentum is set high.

### 3.2 Evaluation

The results of the evaluation show a precision (Equation (1)) of 98.3 % true positives over all classes. Half of the false positive results are due to incorrect or insufficiently accurate localisation. The recall (Equation (2)) value is calculated as 91.7 % for all classes. Here the 360° degree survey prism shows the worst results of the individual targets.

A common method for evaluating object detection is the mean Average Precision (mAP) [21, 30]. This averages the Intersection over Union (IoU) over all classes as well as threshold values. It is therefore a measure of how well the instances were found and classified. The IoU is a measure of the accuracy of the localisation. For the proposed box, the ratio of the intersection and the union with the ground truth is formed. The IoU is the ratio of the two areas intersection and union [21, 30]. Table 3 shows the class AP and mAP for the four different targets.

The exact location of the targets is not relevant to the question. Nevertheless, the results, with a mAP for all classes of 82 % (mean IoU), show the successful detection of the different classes.

In the inference, the network is given a dataset that is completely unknown so far. In order to be able to evaluate the object detection more comprehensively, we have set the value for the minimum confidence to 0.3 in this phase. Exemplary results are shown below in Table 4. The images above illustrate a successful example of each target detection. It should be noted that the representation of the searched objects in the images is ideal, so that high confidence values are achieved. In the second row, both incorrect classifications and results with a small confidence are represented. The bounding boxes determined by the CNN are shown with the respective confidence. It is clear that the bounding boxes do not exactly enclose the targets but are sometimes determined to be too small or too large. Since in this case this is not important for the question,

<table>
<thead>
<tr>
<th>Class (target)</th>
<th>AP for IoU70</th>
<th>AP for IoU80</th>
<th>AP for IoU90</th>
<th>Mean IoU AP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Metal solid sphere</td>
<td>0.98</td>
<td>0.96</td>
<td>0.91</td>
<td>0.93</td>
</tr>
<tr>
<td>360° survey prism</td>
<td>0.79</td>
<td>0.77</td>
<td>0.65</td>
<td>0.72</td>
</tr>
<tr>
<td>Circular prism</td>
<td>0.93</td>
<td>0.88</td>
<td>0.69</td>
<td>0.83</td>
</tr>
<tr>
<td>Black &amp; white target</td>
<td>0.95</td>
<td>0.80</td>
<td>0.59</td>
<td>0.81</td>
</tr>
<tr>
<td>Mean class AP</td>
<td>0.91</td>
<td>0.85</td>
<td>0.71</td>
<td>0.82</td>
</tr>
</tbody>
</table>
Table 4: Examples from inference. The enclosing rectangles are the bounding boxes determined by the CNN. The different colours represent the different classes. cs stands for confidence score.

<table>
<thead>
<tr>
<th>Metal solid sphere</th>
<th>360° survey prism</th>
<th>Circular prism with a removable target plate</th>
<th>Black and white target</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="Image" /></td>
<td><img src="image2.png" alt="Image" /></td>
<td><img src="image3.png" alt="Image" /></td>
<td><img src="image4.png" alt="Image" /></td>
</tr>
<tr>
<td>Correctly classified, cs = 0.84</td>
<td>Correctly classified, cs = 0.81</td>
<td>Correctly classified, cs = 0.85</td>
<td>Correctly classified, cs = 0.71</td>
</tr>
<tr>
<td>Incorrectly classified as 360° prism, cs = 0.58</td>
<td>Correctly classified but with a low cs = 0.34</td>
<td>Incorrectly classified as a black and white target, cs = 0.48</td>
<td>Correctly classified but with a low cs = 0.45</td>
</tr>
</tbody>
</table>

these deviations are acceptable. With the lower threshold value, the metal sphere shows the highest number of false classifications. Since the sphere has a reflective surface, the representation of the sphere in the image changes depending on the environment. Nevertheless, it reaches the highest value with the mAP. However, since the sphere is also very small, the detection is limited to a much smaller distance. It has been shown that in the case of images, especially in the nearer range, the net delivers less favourable results. Often the target is not detected, particularly if it is only partially visible due to the detail.

4 Conclusions

The application for classifying the image scene for the subsequent crosshair detection shows promising approaches. Looking at the results in detail, it becomes apparent that the ML approach shows its advantage in the case of multiple and distinct disturbances of the background. The previous approach using traditional image analysis always calculates the longest lines even if they do not belong to the crosshair. This can lead to an erroneous determination and is only noticeable when checked by the user. The CNN successfully classified these images as unsuitable. By expanding the dataset to include images with uniform backgrounds but insufficient crosshair representation, the classification could be made more robust. Images that were previously still classified as suitable are successfully sorted out in the second trained net. However, deficits are still evident in images with irregular dark backgrounds. Overall, the decision-making process is supported significantly and the user can be relieved of the responsibility for this work step.

An integration of the network into the existing software package is unproblematic due to the implemented image processing software from MVTec. The package would increase significantly in necessary storage space but would remain reasonable (≈200 MB). The resulting time loss for the classification before the detection by means of the already implemented traditional image processing is 200–500 ms per image (2–6 MB, 2456 × 2054 × 8).

The application for object detection of different geodetic targets shows good results. The images are correctly classified with high precision and recall, and the objects are correctly located. Support for a successful application in the measurement process is therefore given. The image-based approach can also be used to identify non-retroreflective targets. Retroreflective targets such as prisms can be differentiated to a certain level and the offset can be applied automatically. A critical factor here is the distance of the target or the sufficient representativeness of the searched object in the image. Since the object detection delivers less accurate results for close-up images and when the target is partially cut off or obscured, one approach would be to expand the dataset specifically for this purpose. It has been shown that the overview camera, due to its specific characteristics, is much more limited in its distance than the telescope camera. From a distance of over 55 m, the images are not usable for training purposes. The dataset could, however, be expanded with images from the other cameras. Investigations into the distance up to which successful object detection can be carried out using these cameras and machine learning are still pending. The size of the target used is also crucial. As the aim is not to detect the target as precisely as possible in the image but only to find it sufficiently precisely to trigger further steps in the measurement
process, the largest possible mAP is not necessary. Here, further investigations could be focused on more precise localisation. The determined bounding boxes could be used as an approximation to search for features in a smaller area using traditional image analyses. In this study, we have limited the possible targets per image to one. This does not correspond to every real-life situation, so there are still more configurations that are possible. Likewise, the minimum confidence value was set at 0.7. This value could be set lower. A generally valid value would first have to be proven in practice.

Author contributions: All the authors have accepted responsibility for the entire content of this submitted manuscript and approved submission.

Research funding: The project is funded by the Carl Zeiss Foundation.

Conflict of interest statement: The authors declare no conflicts of interest regarding this article.

References

16. MVTEC Software GmbH. HALCON operator-referenz 22.05.0.0; 2022.

