Yield Prognosis for the Agrarian Management of Vineyards using Deep Learning for Object Counting

Kai Heinrich[1], Andreas Roth[1], Lukas Breithaupt[1], Björn Möller[1], Johannes Maresch[1]

[1] Technische Universität Dresden
{kai.heinrich,andreas.roth,lukas.breithaupt,
bjoern.moeller,johannes.maresch}@tu-dresden.de

Abstract. In various applications, the counting of objects based on image data plays a pivotal role. In this paper we first conducted a literature review to display the state of the art in counting objects and summarized the results by extracting several important concepts that describe the counting problem as well as the solution. In a second step we applied this knowledge to yield prognosis in vineyards, where we used Deep Learning models to detect the objects. While these methods used in the detection step are state of the art and perform very well, several problems are usually introduced by the constraint of only counting an object once in the counting step. We provide a solution for this common problem by identifying unique objects and tracking them throughout a sequence of images in order to avoid counting objects more than once, resulting in an automated yield prognosis model for vineyards.

Keywords: deep learning, counting, image data, vineyard management

1 Introduction

The forecasting of economical features like profit plays an important role in predictive analytics in the age of digitization. In addition to the service and manufacturing industry, the commodities sectors, especially the field of agrarian management can benefit a lot from disruptive and innovative processes that follow up the digital transformation process [1]. Especially the scientific field of artificial (AI) with its subsidiary field of machine learning (ML) aims to provide a solution to problems that are either very cumbersome for humans to tackle or simply impossible [2]. However, there are still some fields, where artificial intelligence can help to tackle problems that are otherwise cumbersome to tackle.

One of these fields is yield prognosis in the context of agrarian management [3, 4]. The yield prognosis task within vineyard management is often times very labor intensive and subject to errors generated by bad and subjective sampling [5]. In addition to that the manual labor sampling techniques have very poor scaling behavior towards larger vineyards. While there are some automatic approaches using complex lighting

14th International Conference on Wirtschaftsinformatik,
February 24-27, 2019, Siegen, Germany

camera techniques to provide image data for statistical models, they are for the most part even more costly than the actual manual labor sampling process and have to be reapplied every time since the models are subject to various spatial and temporal factors like terrain, season and time of day.

While the problem of decision support systems that excel in the task of predicting profits based on yield prognosis is well established in IS-research, companies often times fail to leverage new analytic tools to harness business values from growing amounts of data [6–8]. While most systems promise improved results in terms of prediction performance, they come with high barriers in terms of application interpretability and required computational power [9, 10]. Therefore, the construction and evaluation of AI technologies like deep learning systems (DLS) in the context of business problems should be considered a vital research interest in IS research [11]. We construct such systems by transferring known algorithms for object counting to the problem of yield prognosis.

The task at hand can therefore be reformulated as finding a model that is able to count agrarian entities like grape berries and whole plants of grape vines. This task adds an additional and important step in the data analysis process: Besides the usual description of data selection, data preparation, modeling and evaluation we add a counting step after the model phase since basic detection is not enough to provide a yield prognosis.

Our goal is to provide several solutions to solve the underlying problems of object detection (modeling phase) and yield prognosis (counting phase) and we therefore employ a two-step-approach: Following a proposed IS research gap by Gordon et al. 2013, we review related work on intelligent systems based on Artificial Neural Networks (ANN) for the task of both object detection and counting the identified objects, propose a systematization of the whole context and secondly evaluate state-of-the-art existing approaches for object recognition in combination with novel approaches for the counting step using image data collected from different vineyards throughout Europe. The paper is therefore structured as follows: In the next section we provide a description of the criteria for our literature review and the subsequent conceptualization of approaches for object counting.

We then proceed to describe the underlying data and analysis process for our data science study in Section 3. In Section 4 the results are provided and discussed with regard to the problem at hand. The last section provides a summary as well as research limitations and an outlook.

2 Preliminaries and related work

The task of counting objects based on image data has a wide variety of application domains, for example, counting cells on microscopic images or the surveillance of large people crowds as well as counting life stock or plants and trees from satellite or drone images [13]. Counting objects on various images is an exhausting and failure heavy process for humans, especially when the data sources contain multiple images (e.g., from video data) that needs to subject to automation for the reason of efficiency [14].

However using intelligent systems to count provides a lot of obstacles to overcome like hidden objects, fuzzy borders between objects or change of perspective [15]. Often times the counting process needs to provide additional information like the number of objects in a certain area (e.g., in air surveillance or agricultural analysis of plant positioning). Loy et al. differentiate between three basic approaches to counting: counting by detection, counting by segmentation and counting by regression [16]. Counting by detection is a two-step approach that first uses localization through object detection (e.g., through the use of a detection model like a convolutional neural network) and then proceeding to counting the detected objects. Counting by segmentation uses multiple images and groups consistent motions throughout the images to estimate the object number. Counting by regression is modeled after the human counting techniques for estimating or “guessing” object numbers without complete enumeration. The growing field of Artificial Intelligence (AI) and Deep learning (DL) enables the application of Artificial Neural networks (ANN), especially convolutional neural networks (CNN) for counting objects. While recent benchmarks suggest that counting by regression outperforms the standard counting by detection it is very cumbersome in the model development phase, since relevant features that help “guessing” the right number have to be crafted by hand for every situation. A CNN requires only labeled training sets and can learn higher order features by itself [17]. A well trained CNN can also be generalized in its architecture to be applied to different objects, e.g., a model that is used to count apples can be trained to count cars without changing the model structure [17]. Related surveys on these topics for both manually generated features and automated feature generation with deep neural networks are provided by [16] and [18] respectively. The surveys provide insights in the effectiveness of the different counting methods. It is suggested that standard counting by detection, while the most precise, fails to live up to the task in cases of images containing a large number of objects. The disadvantage of counting by segmentation is obviously that it cannot be applied to single picture data but needs a consistent stream of pictures with time stamps as they would be generated by videos. While counting by regressions seems to overcome both disadvantages, it often times requires the extraction of low-level features in order to estimate a density function on the domain of those features that reflects the density of objects for every pixel in an image. Integrating then leads to a cumulative function that produces an estimate of the total number of objects in an image [18]. However with the use of DL a certain degree of automation and general applicability can be achieved [19]. Before we evaluate some of the above mentioned approaches for the problem of yield prognosis in the agrarian management of vineyards, we provide a general overview over DL and ANN based algorithms for object counting by conducting a structured literature review.

3 Research Methodology and Literature Review

3.1 Research method

The research goal (RG) of this paper is (1) to provide an extensive state of the art overview of neural network based counting algorithms and (2) provide and evaluate a set of methods to overcome some of the problems that are inherent to counting by detection using data from a case of agrarian management of vineyards.

In order to provide a survey of related literature as formulated in RG (1) we employed a literature review with content analysis [20]. For the data science study formulated in RG (2), we use a study design as suggested by [21]. The structure of our data science study follows the KDDM approach by [22]. It is divided into six basic phases: *i) domain understanding*, *ii) data understanding*, iii) *data preparation*, iv) *modeling*, v) *evaluation*, and vi) *deployment*. Since the system we developed is a non-productive system without deployment we omit this phase from our research process.

As mentioned earlier we add a counting phase after the modeling phase, since using ANN algorithms to detect the objects is not enough. The counting step confronts us with the problem of unique entity counting, meaning that every object should only be counted once in order to provide a valid yield prognosis. We supply solutions to solve both problems – object detection as well as unique counting in our data science study for yield prognosis of vineyards.

For the identification of relevant literature we conducted a search with the terms "count" and either of the keywords "deep learning" or "neural network" in the databases Science Direct, EBSCOhost, IEEE Xplore and arXiv. Both the terms and the database choices cover the area of interest as suggested by RG (1). The search was conducted in a way, that the terms had to be present in at least one of the following meta data of the paper: abstract, title or keywords, where we explicitly defined the search string in a way that the term "count" had to be present in the title as a necessary condition. Initially 321 relevant papers were identified. We then proceeded to apply subjective filtering by screening the title (131) and then the abstract (87) for relevance. After removing duplicates and conducting a backward search we finally yield 99 relevant papers. A forward search based on the 99 papers revealed no additional results. The subjective filter was implemented based on some relevancy checks: (A) counting objects had to be a core topic of the paper, which prevented the inclusion object tracking or object recognition heavy papers; (B) the paper needed to be an original work rather than a survey, so that only papers that describe a method in-depth will be included in the overview; (C) the method was based on image rather than video data; (D) the article had to include neural networks (either standard or deep learning), since that was the focus of our research goal.

From our result set we were able to extract six core concepts that describe the various existing counting methods and their application environments: *architecture, number of objects, density of objects, background dynamics, output, training data* and *type of counting.* In the next section we provide the description of those concepts with the respective results from the literature review.

3.2 Literature Review results

Architecture. This concept distinguishes the counting methods based on extent and type of elements in the data analysis process responsible for counting. First we distinguish systems that only use one step processes (22/99), resulting in a single ANN that can be trained as a whole. Typically those networks were based on CNN architectures (e.g., [23–25]). Those methods typically use local and/or global features (Type of Counting is feature based) to determine the object density per pixel or partial and whole images. Another one step ANN involves classification whether an object is present in a given part of the image or not (Type of counting = Detection based) (e.g., [26]). Secondly there are counting systems that use multi steps that only involve ANN algorithms (16/99). Often times this extraction of partial images and then using it within a regression model (e.g., [27–29]). This machine based feature extraction provides advantages in scenarios with overlapping and strong variations in object size [30]. Other only ANN based multi-step approaches involve sequential models like Long-Term-Short-Term networks (e.g., [31]) or parallel architectures that use multiple networks as base learners while having a final "deciding network" in the second step (e.g. [30, 32, 33]). The last characteristic involves ANNs as well as other methods (61/99). This category acts as a collection for the versatile preprocessing possibilities of ANNs and other image processing methods. ANNs are used to either manipulate images to achieve better results (e.g., scaling or color related features), extract features (e.g. abstract or low level features like edges or texture) and for regression and combination of previous results.

Number of objects. This concept, while it should not be considered isolated, gives us a description of the counting problem at hand that we try to solve using ANN algorithms. The success of various counting methods depend largely on that concept, e.g., for counting of crowds of people we can use detection based methods for a small number of objects but not if we have large crowds present in images. We would then rather use feature based methods. We divided the found literature up into small number of objects ranging from 0-50 (55/99 papers), medium sized 50-200 (21/99 papers), large 200-1000 (14/99) and "outlier" or "extra-large" with the number of objects being larger than 1000 (4/99). Some methods were built to be more flexible and can be applied to various object numbers. In our review we took average numbers of objects based on the used image datasets. We could however not determine such numbers for five publications, since there was no indication regarding the number of objects.

Density of objects. The concept of density is another characteristic attribute of the counting problem and we can distinguish between no overlapping objects (31/99) and overlapping objects (68/99). It was found that for counting problems with considerable overlapping, feature based methods are preferred and more successful in predicting object numbers [13].

Background Dynamic. Another important influence factor found in the literature is the background dynamic which can be distinguished into static (43/99) and dynamic backgrounds (56/99). Static can have multiple meanings in this context: We can assume that over a sequence of images only the foreground changes (e.g., fixed surveillance cameras) where we can ignore the background and remove it via background subtraction methods (e.g., [34–36]) or as another option for static backgrounds we can assume a region of interest as a part of an image that is henceforth declared as foreground (e.g., [37, 38]). The assumption of static background is necessary for some methods to distinguish between foreground and background.

Output. The counting methods found in the literature offer a range of different outputs, depending on the specifics of the problem at hand. In some cases the methods only supply a rough estimate (10/99) without positioning or a total count. This is the case for public transportation surveillance where only a rough percentage estimate is needed, e.g., of how crowded a departure platform is (e.g., [39, 40]), reaching from 0-100%. Some methods output an estimate of the total number of objects in an image (59/99), again without positioning information. To circumvent this problem some approaches divide the image and count the total number of objects by region (8/99). With using overlapping regions, more robust models are generated (e.g., [14, 41]). Other output methods that take positioning into account involve determining a count per pixel by either returning a unique coordinate for each object centroid (2/99) or by using a bounding box that forms a rectangular shape around the object (4/99). Another approach to give an estimate per pixel is using a density function (15/99) as depicted by a heat map in Figure 1, where a higher pixel density is expressed by red color whereas low densities are expressed by blue color (e.g., [13, 25, 42]).

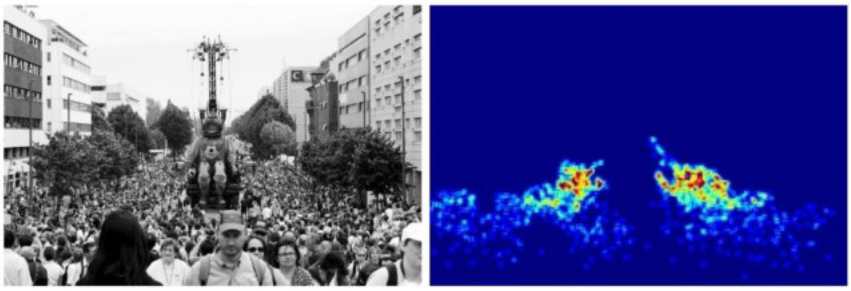

Figure 1. Example of density based count per pixel method ([28])

Training Data. The different options of available training data can be characterized by the effort of annotation of the unlabeled data. The first annotation tier with the lowest effort is given by annotating the total count (38/99). This is particularly useful when we only need an estimate or the total number as output without positioning information. For the purpose of annotating with positioning information we need the centroid coordinate (35/99), where every object is manually annotated with a dot in the center. This type of training data can be used for all output types since some outputs require the image to be divided up into regions, which in turn requires positioning information such as centroid coordinates (e.g., [28]). It was also found that this kind of data is needed in order to create density maps using Gaussian filter techniques (e.g. [30]). There are also approaches that do not require the annotation of training data at all (19/99) like segmentation based counting (e.g., [43, 44]) or detection based counting which instead of annotations needs example images of the objects that are to be counted (e.g., [44]).

Type of counting. The last concept as extracted from the literature review gives an insight on how the different approaches work. The first option is to use segmentation methods (15/99). After background and noise removal, which are essential preprocessing steps for this kind of approach, every foreground segment is declared as one object instance (e.g., [43]). The foreground/background segmentation is either conducted by a classification algorithm or by a threshold transforming images into binary black and white images with white pixels representing foreground objects [45]. The detection based methods (15/99) use a multi-step approach where objects are detected first and then localized and counted afterwards. Depending on the task at hand the detector is either trained to recognize the whole object (e.g. [44]) or only parts of it (e.g., only the head of a human body). The detectors are trained with single image training data to determine whether the object of interest is present or not. However a shortcoming of those approaches is that they cannot successfully be used with strong overlapping and large number of objects [16, 18]. Feature based methods (69/99) can circumvent that downside by extracting local and/or global features of the image. This can either be done by manually crafting features (e.g., [46]) or to use annotated data to train feature extractors (e.g., [23, 27]). Manually extracted features include shape, color, area, diameter, roughness of texture (e.g. [47]) or even additional meta annotations like perspective or weather data [48]. After feature extraction common machine learning algorithms like SVM, linear regression or ANNs are used to predict the desired output. In the next section we introduce a data science study that describes a data analysis process to provide an estimate for counting vines and grapes in order to produce a yield prediction to support decisions in vineyard management.

4 Data Science Study

4.1 Domain and Data Understanding

The management task in the context of agrarian management of vineyards is faced coherent spatial variability in the production systems, making crop and yield vigour an essential information [49]. Temporal variations of crop stability lead to unpredictable and nonstable yields, so that even the information of current yield during any phase of the crop development is vital.

In our data science study we gathered image data from three different vineyards in Germany at three different stages of the crop process (early, mid and late). The data gathering process was designed to be as cost efficient as possible, utilizing already existing equipment and avoiding complex lighting procedures and other high-cost image capturing in order to determine if a system can be designed that solves the counting task with only average to weak image quality data. We therefore combined already available mobile harvesting equipment like tractors with image capturing devices like full-HD cameras that generated a sequence of single images along a field of vine crops. We gathered three hours of video material in HD quality[1] and some Ultra-

[1] Taken with a GoPro Hero5 that was attached to the harvesting equipment

HD examples for testing purposes. Some important constraints when facing our task are a generalizable model that can predict yield under different conditions like weather, time of day and vineyard structure (e.g., slope or soil structure) and the necessity of avoiding to count an object more than once.

Since we can only use image data to train the model we transformed the video material into approximately 50 GB of image data sequences. We then judged the quality of the resulting images and especially the information content and removed poor quality images and images with no information content. Every image was captured from a frontal perspective facing the vines and we excluded other perspectives. We also excluded special vine crop structures like blue protection grids in order to keep generalizability (those vines have the same basic structure as other vine crops and are therefore detected just fine later on).

We can describe our counting problem in terms of the concepts extracted earlier in the literature review as a *small number of objects problem (0-50)* with *clear distinction* between the objects in terms of object density and for the most part *static backgrounds* with only small variations (e.g., cloud movement). We employ a *multi-step approach combining ANNs*. Since the object number is quiet high and we have some overlapping on the grapes, we use feature based detection. However we do not identify the image features ourselves but instead minimize the feature selection process by letting state-of-the-art image detection networks engineer the features, which further reduces manual labor on the part of the vineyard management. The concepts of output, training data and type of counting are given in the subsequent sections.

4.2 Preprocessing

For the labeling process we used the concept of *bounding boxes* for the training data as they are a supported format by deep learning frameworks like TensorFlow[2] and are our designated form of output for the modeling stages. We also labeled metal and wood rods that are occasionally used in vineyards in order to avoid confusing the algorithm. In other preprocessing steps we applied noise removal and feature extraction to highlight different parts of an image using thresholds and the background/foreground distinction resulting in black and white images, picturing vines with white pixels. For noise removal and isolation of objects of interest we further applied erosion on the transformed b/w images utilizing the OpenCV programming library and added Canny Edge Detection to better distinguish grape vines from the rest.[50]. Figure 2 gives a visualization of those preprocessing steps.

[2] https://www.tensorflow.org/

Figure 2. Selection criteria for input image data

For evaluation, we divided the data set into 70/30 train and validation data partitions.

4.3 Modeling

Given our counting task problem and the results of our literature review, we decided on a *detection based* type of counting by using automated feature extraction to minimize the manual effort in model building. There exist a lot of algorithms for efficient object detection and TensorFlow provides a collection of some of the state-of-the-art algorithms for that task [51]. We utilized the three common architectures for object detection ANNs: Single Shot Multi-Box Detector (SSD), Faster Region-Convolutional Neural Network (Faster R-CNN) and Region-based Fully Convolutional Networks (R-FCN). As a basis for our model training we used models that were pre-trained on the common benchmarking datasets: COCO (common objects in context), KITTI (images that capture traffic situations) and OID (an open Image dataset with 9 million images[3]).

We then continued training for the pre-trained models based on our training and test datasets. The results of the detection step in terms of model loss are given in Table 1. For evaluation purpose we compared the ground truth labeling bounding boxes with the model output boxes and used the mean average precision (mAP) as a metric [52]. In order for the comparison to be conclusive we use the degree of intersection between the two bounding boxes (ground truth vs. model output), also defined as Intersection over Union (IoU). The minimum threshold suggested by [52] is declared as an IoU of 50%. Since the original threshold was meant for more than 100 image object classes and we only have three classes (vine, metalstick, woodenstick), so we used an IoU of 70%.

Table 1. Object Detection Model Results

Model Architecture/ pre-training	Average Precision			Mean AP@0.5 IoU
	Vine	Wood	Metal	
F.R-CNN/coco	0.9887	0.9995	0.9992	0.9962
F.R.-CNN/kitti	0.8775	0.9081	0.9523	0.9126
F.R.-CNN/oid	0.9839	0.9991	0.9702	0.9847
SSD/coco	0.6027	0.6910	0.4433	0.5790
R-FCN/coco	0.9729	0.9982	0.9997	0.9910

[3] https://github.com/openimages/dataset

We can see that the task of vine detection executed successfully by all models except the SSD. For grape detection we therefore only used R-FCN and F.R-CNN models. Detecting single grapes was a nearly impossible task, since even with filtering the labeling of berries by humans was cumbersome and not always successful. We obtained average mAP values ranging from 0.2670 to 0.822, with the highest values only being reached when using some UHD quality images. Since obtaining and analyzing UHD images for a whole vineyard is beyond our economical reasonable goal definition of automated, cheap counting methods, we decided to cluster the berries up to grape vines and obtained mAP values ranging from 0.7576 to 0.9811 using an even higher IoU of 75%. The F.R-CNN/oid performed best with detecting grape vines.

4.4 Counting

As mentioned before we encounter several problems despite the successful detection of vines and grapes, since we cannot allow to count objects more than once. Using sequential image data we can define a simple predecessor – successor tracking process by giving an ID to a random box starting in frame 0 and then identifying that box in the next frame by drawing the same box from the previous frame and calculating overlapping percentage to all other boxes in the frame. The box with the most overlapping is called the successor and gets the same item ID. This creates a chain and identifies a single vine throughout the sequence. The process resets when there is no successor defined (when the end of the chain is reached) anymore and another box from frame 0 get a new ID. If all boxes already have an ID the process starts at frame 1 and so on, until all the vines have unique IDs. Sometimes the camera moves to fast or the camera reaches the end of the chain, so that we implemented a distance check that calculates the mean distance between two boxes. Whenever this distance is larger than the threshold of 150, no successor is defined, which lets the tracking process start again as explained above. The process is simplified and depicted in Figure 3 with the successor frames shown as red rectangles in frame n+1. A problem when using this method arises when we have (a) either missing vines in reality (e.g. vines that were removed because of crop sickness) or we (b) failed to detect a vine because it was hidden behind some other object (e.g. leaves or other plants).

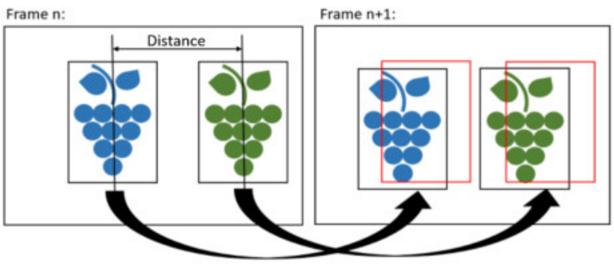

Figure 3. Tracking with overlapping and average distance between objects

In this case our method would not determine a successor object and counting would stop. But since we can track the vine throughout a sequence of images, we can determine the position the vine would have been at in that particular frame where case (a) or (b) applies. In case (a) we applied a 1.5 times threshold to the average distance measure as depicted in Figure 4 between two different IDs. For example if we calculate an average distance between two different vines (=different IDs) of 261 pixels, the threshold would be 391.5 pixels, resulting in the declaration of a missing vine whenever the distance between two bounding boxes exceeds that threshold. In case (b) we can correct the position where the successor should be by adding the average predecessor-successor difference to the previous position. If the object is out of bounce regarding to the frame coordinates we have simply reached the end of the row, otherwise we have imputed a vine which was not detected by our DL algorithm in the first place because of reason (b). This way we can solve the unique counting problem and also detect missing vines to give feedback to vineyard managers. This can have positive impacts especially when combined with agrarian management information like logs as to why vines were removed, so that we can prevent sickness from spreading or isolating certain crops in the development process.

We applied our process to every row we gathered video material from and the average deviation between model count and true vine count was 0.12 per row, with extreme values of 0 and 2. The average deviation for the count of grape vines per row was 1.87 with extreme values of 0 and 4. The extreme values can be explained by bad detection model performance on some objects, where the distance based algorithm just assumed there is either a missing vine or the end of row is reached. It is important to note that the counting process performance itself does not result in an error state itself, only when the object detection framework fails to classify an object. In comparison with other mechanics like motion tracking [25] or neural network based tracking [53] this combination is easier to use in the vine counting scenario as we described by using the extracted concepts from the literature review: medium number of objects, no overlapping, low density which is not the case in most benchmark scenarios of tracking in literature, where often times people are tracked in large crowds, facing problems of blur, high density and large amounts of overlapping, which requires far more advanced approaches [54]. However, in this case, it is sufficient in terms of the economic application to use a simple distance based method, since in comparison to the people tracking benchmarks we have a scenario with only slight perspective changes, distinguishable or static background and the overall scene is the same in every vineyard, apart from some details that have to be learned by the detection model.

5 Discussion & Outlook

We provided a literature review to describe both: the counting problem itself and the solution using various concepts like number of objects or type of counting. We then implemented a data science study to count objects like vines and grapes based on vineyard image data in order to provide a yield prognosis. While the detection of vines and grapes was very successful, the models failed to detect the single berries for non-

UHD picture material. We also provided a tracking solution for the counting step, to fulfill the constraint of unique object counting. However we only used a simple approach here that was based on subjective thresholds that might be subject to change on other vineyard architectures. Possibilities to circumvent this problem is the application of motion tracking straight onto the video material. While we made sure we had GPS data while gathering the image material we did not utilize it, which could be done in a next step and combined with aerial footage to map out the vineyard and provide more useful information like disease spreading factors to vineyard management, all of which would be automated and would not require human intervention once fully implemented. The process and the ideas provided in this paper are generally very robust to change of environment, especially the pre-trained networks can work on similar image data or can be re-trained in only a short amount of time, so that this process could be generalized onto various agrarian management situations were a similar counting problem structure can be found in terms of number of objects, density and background dynamics.

References

1. Shen, F., Chao, J., Zhao, J.: Forecasting Exchange Rate Using Deep Belief Networks and Conjugate Gradient Method. Neurocomput. 167, 243–253 (2015).
2. Goodfellow, I., Bengio, Y., Courville, A.: Deep learning. MIT Press, Cambridge, Massachusetts (2016).
3. Font, D., Tresanchez, M., Martínez, D., Moreno, J., Clotet, E., Palacín, J.: Vineyard yield estimation based on the analysis of high resolution images obtained with artificial illumination at night. Sensors. 15, 8284–8301 (2015).
4. Fernandes, J.L.E., Nelson Francisco;Dalla Mora Esquerdo, Julio Cesar: Sugarcane yield prediction in Brazil using NDVI time series and neural networks ensemble. INTERNATIONAL JOURNAL OF REMOTE SENSING. 38, 4631–4644 (2017).
5. Nuske, S., Wilshusen, K., Achar, S., Yoder, L., Narasimhan, S., Singh, S.: Automated visual yield estimation in vineyards. Journal of Field Robotics. 31, 837–860 (2014).
6. Chen, H., Chiang, R.H.L., Storey, V.C.: Business Intelligence and Analytics: From Big Data to Big Impact. MIS Q. 36, 1165–1188 (2012).
7. Sharma, R., Mithas, S., Kankanhalli, A.: Transforming decision-making processes: a research agenda for understanding the impact of business analytics on organisations. European Journal of Information Systems. 23, 433–441 (2014).
8. Shmueli, G., Koppius, O.R.: Predictive analytics in information systems research. Mis Quarterly. 553–572 (2011).
9. Arel, I., Rose, D.C., Karnowski, T.P.: Deep machine learning-a new frontier in artificial intelligence research [research frontier]. IEEE computational intelligence magazine. 5, 13–18 (2010).
10. Larochelle, H., Erhan, D., Courville, A., Bergstra, J., Bengio, Y.: An empirical evaluation of deep architectures on problems with many factors of variation. In: Proceedings of the 24th international conference on Machine learning. pp. 473–480. ACM (2007).
11. Agarwal, R., Dhar, V.: Editorial—Big Data, Data Science, and Analytics: The Opportunity and Challenge for IS Research. Information Systems Research. 25, 443–448 (2014).

12. Gordon, S., Blake, R., Shankaranarayanan, G.: Case-based research in information systems: Gaps and trends. JITTA: Journal of Information Technology Theory and Application. 14, 47 (2013).
13. Lempitsky, V., Zisserman, A.: Learning to count objects in images. In: Advances in Neural Information Processing Systems. pp. 1324–1332 (2010).
14. Cohen, J.P., Boucher, G., Glastonbury, C.A., Lo, H.Z., Bengio, Y.: Count-ception: Counting by Fully Convolutional Redundant Counting. arXiv:1703.08710 [cs, stat]. (2017).
15. Zhang, C.: Design of Human Capability Maturity Analysis System Online P-CMM Model. Presented at the December (2015).
16. Loy, C.C., Chen, K., Gong, S., Xiang, T.: Crowd Counting and Profiling: Methodology and Evaluation. In: Ali, S., Nishino, K., Manocha, D., and Shah, M. (eds.) Modeling, Simulation and Visual Analysis of Crowds: A Multidisciplinary Perspective. pp. 347–382. Springer New York, New York, NY (2013).
17. Seguí, S., Pujol, O., Vitrià, J.: Learning to count with deep object features. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). pp. 90–96 (2015).
18. Sindagi, V.A., Patel, V.M.: A survey of recent advances in CNN-based single image crowd counting and density estimation. Pattern Recognition Letters. (2017).
19. Goodfellow, I., Lee, H., Le, Q.V., Saxe, A., Ng, A.Y.: Measuring invariances in deep networks. In: Advances in neural information processing systems. pp. 646–654 (2009).
20. Webster, J., Watson, R.T.: Analyzing the past to prepare for the future: Writing a. MIS quarterly. 26, 13–23 (2002).
21. Müller, O., Junglas, I., vom Brocke, J., Debortoli, S.: Utilizing big data analytics for information systems research: challenges, promises and guidelines. European Journal of Information Systems. 25, 289–302 (2016).
22. Kurgan, L.A., Musilek, P.: A survey of Knowledge Discovery and Data Mining process models. The Knowledge Engineering Review. 21, 1–24 (2006).
23. Capobianco, S., Marinai, S.: Deep neural networks for record counting in historical handwritten documents. Pattern Recognition Letters. (2017).
24. Hao Lu, Zhiguo Cao, Yang Xiao, Bohan Zhuang, Chunhua Shen: TasselNet: counting maize tassels in the wild via local counts regression network. Plant Methods. 13, 1–17 (2017).
25. Kang, D., Ma, Z., Chan, A.B.: Beyond Counting: Comparisons of Density Maps for Crowd Analysis Tasks - Counting, Detection, and Tracking. arXiv:1705.10118 [cs]. (2017).
26. Achler, T., Vural, D.C., Amir, E.: Counting objects with biologically inspired regulatory-feedback networks. In: 2009 International Joint Conference on Neural Networks. pp. 36–40 (2009).
27. Aich, S., Stavness, I.: Leaf Counting with Deep Convolutional and Deconvolutional Networks. arXiv:1708.07570 [cs]. (2017).
28. Marsden, M., McGuinness, K., Little, S., Keogh, C.E., O'Connor, N.E.: People, Penguins and Petri Dishes: Adapting Object Counting Models To New Visual Domains And Object Types Without Forgetting. arXiv:1711.05586 [cs]. (2017).
29. Tang, S., Pan, Z., Zhou, X.: Low-Rank and Sparse Based Deep-Fusion Convolutional Neural Network for Crowd Counting. Mathematical Problems in Engineering. 1–11 (2017).
30. Boominathan, L., Kruthiventi, S.S.S., Babu, R.V.: CrowdNet: A Deep Convolutional Network for Dense Crowd Counting. arXiv:1608.06197 [cs]. (2016).
31. Yao, H., Han, K., Wan, W., Hou, L.: Deep Spatial Regression Model for Image Crowd Counting. arXiv:1710.09757 [cs]. (2017).
32. Onoro-Rubio, D., López-Sastre, R.J.: Towards perspective-free object counting with deep learning. In: European Conference on Computer Vision. pp. 615–629. Springer (2016).

33. Zhang, Y., Zhou, D., Chen, S., Gao, S., Ma, Y.: Single-Image Crowd Counting via Multi-Column Convolutional Neural Network. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 589–597 (2016).
34. Amin, I.J., Taylor, A.J., Junejo, F., Al-Habaibeh, A., Parkin, R.M.: Automated people-counting by using low-resolution infrared and visual cameras. Measurement. 41, 589–599 (2008).
35. French, G., Fisher, M., Mackiewicz, M., Needle, C.: Convolutional Neural Networks for Counting Fish in Fisheries Surveillance Video. Presented at the (2015).
36. Yoshinaga, S., Shimada, A., Taniguchi, R.: Real-time people counting using blob descriptor. Procedia - Social and Behavioral Sciences. 2, 143–152 (2010).
37. Laki, A., Rattalino, I., Corinto, F., Iván, K., Demarchi, D., Civera, P.: An integrated LOC hydrodynamic focuser with a CNN-based camera system for cell counting application. In: 2011 IEEE Biomedical Circuits and Systems Conference (BioCAS). pp. 301–304 (2011).
38. Hu, X., Zheng, H., Chen, Y., Chen, L.: Dense crowd counting based on perspective weight model using a fisheye camera. Optik - International Journal for Light and Electron Optics. 126, 123–130 (2015).
39. Cho, S.-Y., Chow, T.W.S., Leung, C.-T.: A neural-based crowd estimation by hybrid global learning algorithm. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics). 29, 535–541 (1999).
40. Chow, T.W.S., Cho, S.-Y.: Industrial neural vision system for underground railway station platform surveillance. Advanced Engineering Informatics. 16, 73–83 (2002).
41. Han, K., Wan, W., Yao, H., Hou, L.: Image Crowd Counting Using Convolutional Neural Network and Markov Random Field. arXiv:1706.03686 [cs]. (2017).
42. Zeng, L., Xu, X., Cai, B., Qiu, S., Zhang, T.: Multi-scale Convolutional Neural Networks for Crowd Counting. arXiv:1702.02359 [cs]. (2017).
43. Bertucco, L., Nunnari, G., Randieri, C., Rizza, V., Sacco, A.: A cellular neural network based system for cell counting in culture of biological cells. In: Proceedings of the 1998 IEEE International Conference on Control Applications (Cat. No.98CH36104). pp. 341–345 vol.1 (1998).
44. Biswas, D., Hongbo Su, Wang, C., Blankenship, J., Stevanovic, A.: An Automatic Car Counting System Using OverFeat Framework. Sensors (14248220). 17, 1–13 (2017).
45. Schofield, A.J., Mehta, P.A., Stonham, T.J.: A system for counting people in video images using neural networks to identify the background scene. Pattern Recognition. 29, 1421–1428 (1996).
46. Embleton, K.V., Gibson, C.E., Heaney, S.I.: Automated counting of phytoplankton by pattern recognition: a comparison with a manual counting method. J Plankton Res. 25, 669–681 (2003).
47. Lal, S., Behera, S.K., Sethy, P.K., Rath, A.K.: Identification and counting of mature apple fruit based on BP feed forward neural network. In: 2017 Third International Conference on Sensing, Signal Processing and Security (ICSSS). pp. 361–368 (2017).
48. Nazeer, M., Wong, M.S., Nichol, J.E.: A new approach for the estimation of phytoplankton cell counts associated with algal blooms. Science of The Total Environment. 590–591, 125–138 (2017).
49. Taylor, J., Tisseyre, B., Bramley, R., Reid, A., Stafford, J.: A comparison of the spatial variability of vineyard yield in European and Australian production systems. Precision agriculture. 5, 907–914 (2005).
50. Bradski, G., Kaehler, A.: Learning OpenCV: Computer Vision in C++ with the OpenCV Library. O'Reilly Media, Inc. (2013).

51. Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M.: Tensorflow: a system for large-scale machine learning. In: OSDI. pp. 265–283 (2016).
52. Everingham, M., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. International journal of computer vision. 88, 303–338 (2010).
53. Nam, H., Han, B.: Learning multi-domain convolutional neural networks for visual tracking. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4293–4302 (2016).
54. Chan, A.B., Liang, Z.-S.J., Vasconcelos, N.: Privacy preserving crowd monitoring: Counting people without people models or tracking. In: 2008 IEEE Conference on Computer Vision and Pattern Recognition. pp. 1–7 (2008).