A Robust Image Object Recognition Method Using RCNN and Big Data

Radhamadhab Dalai, PhD Scholar
Department of computer science and Engineering
BIT Mesra

Kishore Kumar Senapati, Assistant Professor
Department of computer science and Engineering
BIT Mesra

ABSTRACT

This paper presents a novel approach to object detection using deep convolutional neural networks and Big Data. The aim is to build an accurate, fast and reliable object detection system, which is a vital element of an autonomous robotic vision platform; it is a key element for object estimation and automated event based systems. Recent work in deep neural networks has led to the development of a state-of-the-art object detector termed Faster Region-based CNN (Mask R-CNN). This model, through transfer learning, for the task of object detection using imagery obtained from two modalities: color (RGB) and Near-Infrared (NIR). Early and late fusion methods are explored for combining the multi-modal (RGB and NIR) information. This leads to a novel multi-modal masked R-CNN model, which achieves state-of-the-art results compared to prior work with the F1 score, which takes into account both precision and recall performances improving from 0.781 to 0.812 for the detection of solid metal objects. In addition to improved accuracy, this approach is also much quicker to deploy for new objects, as it requires bounding box annotation rather than pixel-level annotation (annotating bounding boxes is approximately an order of magnitude quicker to perform). The model is retrained to perform the detection of four types of metal objects, with the entire process taking four hours to annotate and train the new model per solid block. A key benefit of Deep Learning is the analysis and learning of massive amounts of unsupervised data, making it a valuable tool for Big Data Analytics where raw data is largely unlabeled and un-categorized. In the present study, we explore how Deep Learning can be utilized for addressing some important problems in Big Data Analytics, including extracting complex patterns from massive volumes of data, semantic indexing, data tagging, fast information retrieval, and simplifying discriminative tasks.

General Terms
Object detection, CNN, Big Data

Keywords
R-CNN, Mask RCNN, Deep Learning, bounding box annotation.

1. INTRODUCTION

The general focus of machine learning is the representation of the input data and generalization of the learnt patterns for use on future unseen data. The goodness of the data representation has a large impact on the performance of machine learners on the data: a poor data representation is likely to reduce the performance of even an advanced, complex machine learner, while a good data representation can lead to high performance for a relatively simpler machine learner. Thus, feature engineering, which focuses on constructing features and data representations from raw data [1], is an important element of machine learning. Feature engineering consumes a large portion of the effort in a machine learning task, and is typically quite domain specific and involves considerable human input. For example, the Histogram of Oriented Gradients (HOG) and Scale Invariant Feature Transform (SIFT) are popular feature engineering algorithms developed specifically for the computer vision domain. Performing feature engineering in a more automated and general fashion would be a major breakthrough in machine learning as this would allow practitioners to automatically extract such features without direct human input.

Deep Learning algorithms are one promising avenue of research into the automated extraction of complex data representations (features) at high levels of abstraction. Such algorithms develop a layered, hierarchical architecture of learning and representing data, where higher-level (more abstract) features are defined in terms of lower-level (less abstract) features. The hierarchical learning architecture of Deep Learning algorithms is motivated by artificial intelligence emulating the deep, layered learning process of the primary sensorial areas of the neocortex in the human brain, which automatically extracts features and abstractions from the underlying data [4]-[6]. Deep Learning algorithms are quite beneficial when dealing with learning from large amounts of unsupervised data, and typically learn data representations in a greedy layer-wise fashion [7],[8]. Empirical studies have demonstrated that data representations obtained from stacking up non-linear feature extractors (as in Deep Learning) often yield better machine learning results, e.g., improved classification modeling [9], better quality of generated samples by generative probabilistic models [10], and the invariant property of data representations [11]. Deep Learning solutions have yielded outstanding results in different machine learning applications, including speech recognition [12]-[16], computer vision [7],[8],[14], and natural language processing. A more detailed overview of Deep Learning is presented in Section “Deep learning in data mining and machine learning”.

Big Data represents the general realm of problems and techniques used for application domains that collect and maintain massive volumes of raw data for domain-specific data analysis. Modern data-intensive technologies as well as increased computational and data storage resources have contributed heavily to the development of Big Data science [21]. Technology based companies such as Google, Yahoo, Microsoft, and Amazon have collected and maintained data that is measured in exabyte proportions or larger. Moreover, social media organizations such as Facebook, YouTube, and Twitter have billions of users that constantly generate a very large quantity of data. Various organizations have invested in

developing products using Big Data Analytics to addressing their monitoring, experimentation, data analysis, simulations, and other knowledge and business needs [22], making it a central topic in data science research.

Mining and extracting meaningful patterns from massive input data for decision-making, prediction, and other inferencing is at the core of Big Data Analytics. In addition to analyzing massive volumes of data, Big Data Analytics poses other unique challenges for machine learning and data analysis, including format variation of the raw data, fast-moving streaming data, trustworthiness of the data analysis, highly distributed input sources, noisy and poor quality data, high dimensionality, scalability of algorithms, imbalanced input data, unsupervised and un-categorized data, limited supervised/labeled data, etc. Adequate data storage, data indexing/tagging, and fast information retrieval are other key problems in Big Data Analytics. Consequently, innovative data analysis and data management solutions are warranted when working with Big Data. For example, in a recent work we examined the high-dimensionality of bioinformatics domain data and investigated feature selection techniques to address the problem [23]. A more detailed overview of Big Data Analytics is presented in Section "Big data analytics".

The knowledge learnt from (and made available by) Deep Learning algorithms has been largely untapped in the context of Big Data Analytics. Certain Big Data domains, such as computer vision [7] and speech recognition [13], have seen the application of Deep Learning largely to improve classification modeling results. The ability of Deep Learning to extract high-level, complex abstractions and data representations from large volumes of data, especially unsupervised data, makes it attractive as a valuable tool for Big Data Analtyics. More specifically, Big Data problems such as semantic indexing, data tagging, fast information retrieval, and discriminative modeling can be better addressed with the aid of Deep Learning. More traditional machine learning and feature engineering algorithms are not efficient enough to extract the complex and non-linear patterns generally observed in Big Data. By extracting such features, Deep Learning enables the use of relatively simpler linear models for Big Data analysis tasks, such as classification and prediction, which is important when developing models to deal with the scale of Big Data. The novelty of this study is that it explores the application of Deep Learning algorithms for key problems in Big Data Analytics, motivating further targeted research by experts in these twofields.

The paper focuses on two key topics: (1) how Deep Learning can assist with specific problems in Big Data Analytics, and (2) how specific areas of Deep Learning can be improved to reflect certain challenges associated with Big Data Analytics. With respect to the first topic, we explore the application of Deep Learning for specific Big Data Analytics, including learning from massive volumes of data, semantic indexing, discriminative tasks, and data tagging. Our investigation regarding the second topic focuses on specific challenges Deep Learning faces due to existing problems in Big Data Analytics, including learning from streaming data, dealing with high dimensionality of data, scalability of models, and distributed and parallel computing. We conclude by identifying important future areas needing innovation in Deep Learning for Big Data Analytics, including data sampling for generating useful high-level abstractions, domain (data distribution) adaption, defining criteria for extracting good data representations for discriminative and indexing tasks, semi-supervised learning, and active learning.

The remainder of the paper is structured as follows: Section "Deep learning in data mining and machine learning" presents an overview of Deep Learning for data analysis in data mining and machine learning; Section "Big data analytics" presents an overview of Big Data Analytics, including key characteristics of Big Data and identifying specific data analysis problems faced in Big Data Analytics; Section "Applications of deep learning in big data analytics" presents a targeted survey of works investigating Deep Learning based solutions for data analysis, and discusses how Deep Learning can be applied for Big Data Analytics problems; Section "Deep learning challenges in big data analytics" discusses some challenges faced by Deep Learning experts due to specific data analysis needs of Big Data; Section "Future work on deep learning in big data analytics" presents our insights into further works that are necessary for extending the application of Deep Learning in Big Data, and poses important questions to domain experts; and in Section "Conclusion" we reiterate the focus of the paper and summarize the work presented.

The main concept in deep leaning algorithms is automating the extraction of representations (abstractions) from the data [1],[4],[5]. Deep learning algorithms use a huge amount of unsupervised data to automatically extract complex representation. These algorithms are largely motivated by the field of artificial intelligence, which has the general goal of emulating the human brain's ability to observe, analyze, learn, and make decisions, especially for extremely complex problems. Work pertaining to these complex challenges has been a key motivation behind Deep Learning algorithms which strive to emulate the hierarchical learning approach of the human brain. Models based on shallow learning architectures such as decision trees, support vector machines, and case-based reasoning may fall short when attempting to extract useful information from complex structures and relationships in the input corpus. In contrast, Deep Learning architectures have the capability to generalize in non-local and global ways, generating learning patterns and relationships beyond immediate neighbors in the data [4]. Deep learning is in fact an important step toward artificial intelligence. It not only provides complex representations of data which are suitable for AI tasks but also makes the machines independent of human knowledge which is the ultimate goal of AI. It extracts representations directly from unsupervised data without human interference.

A key concept underlying Deep Learning methods is distributed representations of the data, in which a large number of possible configurations of the abstract features of the input data are feasible, allowing for a compact representation of each sample and leading to a richer generalization. The number of possible configurations is exponentially related to the number of extracted abstract features. Noting that the observed data was generated through interactions of several known/unknown factors, and thus when a data pattern is obtained through some configurations of learnt factors, additional (unseen) data patterns can likely be described through new configurations of the learnt factors and patterns[5],[24]. Compared to learning based on local generalizations, the number of patterns that can be obtained using a distributed representation scales quickly with the number of learnt factors.

Deep learning algorithms lead to abstract representations because more abstract representations are often constructed based on less abstract ones. An important advantage of more abstract representations is that they can be invariant to the local changes in the input data. Learning such invariant features is an ongoing major goal in pattern recognition (for example learning features that are invariant to the face orientation in a face recognition task). Beyond being invariant such representations can also disentangle the factors of variation in data. The real data used in AI-related tasks mostly arise from complicated interactions of many sources. For example an image is composed of different sources of

variations such a light, object shapes, and object materials. The abstract representations provided by deep learning algorithms can separate the different sources of variations in data.

Deep learning algorithms are actually Deep architectures of consecutive layers. Each layer applies a nonlinear transformation on its input and provides a representation in its output. The objective is to learn a complicated and abstract representation of the data in a hierarchical manner by passing the data through multiple transformation layers. The sensory data (for example pixels in an image) is fed to the first layer. Consequently the output of each layer is provided as input to its next layer.

Stacking up the nonlinear transformation layers is the basic idea in deep learning algorithms. The more layers the data goes through in the deep architecture, the more complicated the nonlinear transformations which are constructed. These transformations represent the data, so Deep Learning can be considered as special case of representation learning algorithms which learn representations of the data in a Deep Architecture with multiple levels of representations. The achieved final representation is a highly non-linear function of the input data.

It is important to note that the transformations in the layers of deep architecture are non-linear transformations which try to extract underlying explanatory factors in the data. One cannot use a linear transformation like PCA as the transformation algorithms in the layers of the deep structure because the compositions of linear transformations yield another linear transformation. Therefore, there would be no point in having a deep architecture. For example by providing some face images to the Deep Learning algorithm, at the first layer it can learn the edges in different orientations; in the second layer it composes these edges to learn more complex features like different parts of a face such as lips, noses and eyes. In the third layer it composes these features to learn even more complex feature like face shapes of different persons. These final representations can be used as feature in applications of face recognition. This example is provided to simply explain in an understandable way how a deep learning algorithm finds more abstract and complicated representations of data by composing representations acquired in a hierarchical architecture. However, it must be considered that deep learning algorithms do not necessarily attempt to construct a pre-defined sequence of representations at each layer (such as edges, eyes, faces), but instead more generally perform non-linear transformations in different layers. These transformations tend to disentangle factors of variations in data. Translating this concept to appropriate training criteria is still one of the main open questions in deep learning algorithms [5].

The final representation of data constructed by the deep learning algorithm (output of the final layer) provides useful information from the data which can be used as features in building classifiers, or even can be used for data indexing and other applications which are more efficient when using abstract representations of data rather than high dimensional sensory data.

Learning the parameters in a deep architecture is a difficult optimization task, such as learning the parameters in neural networks with many hidden layers. In 2006 Hinton proposed learning deep architectures in an unsupervised greedy layer-wise learning manner [7]. At the beginning the sensory data is fed as learning data to the first layer. The first layer is then trained based on this data, and the output of the first layer (the first level of learnt representations) is provided as learning data to the second layer. Such iteration is done until the desired number of layers is obtained. At this point the deep network is trained. The representations learnt on the last layer can be used for different tasks. If the task is a classification task usually another supervised layer is put on top of the last layer and its parameters are learnt (either randomly or by using supervised data and keeping the rest of the network fixed). At the end the whole network is fine-tuned by providing supervised data to it.

Here we explain two fundamental building blocks, unsupervised single layer learning algorithms which are used to construct deeper models: Autoencoders and Restricted Boltzmann Machines (RBMs). These are often employed in tandem to construct stacked Autoencoders [8], [6] and Deep belief networks [7], which are constructed by stacking up Autoencoders and Restricted Boltzmann Machines respectively. Autoencoders, also called autoassociators [7], are networks constructed of 3 layers: input, hidden and output. Autoencoders try to learn some representations of the input in the hidden layer in a way that makes it possible to reconstruct the input in the output layer based on these intermediate representations. Thus, the target output is the input itself. A basic Autoencoder learns its parameters by minimizing the reconstruction error. This minimization is usually done by stochastic gradient descent (much like what is done in Multilayer Perceptron). If the hidden layer is linear and the mean squared error is used as the reconstruction criteria, then the Autoencoder will learn the first k principle components of the data. Alternative strategies are proposed to make Autoencoders nonlinear which are appropriate to build deep networks as well as to extract meaningful representations of data rather than performing just as a dimensionality reduction method. They have called these methods "regularized Autoencoders" in [5], and we refer an interested reader to that paper for more details on algorithms.

Another unsupervised single layer learning algorithm which is used as a building block in constructing Deep Belief Networks is the Restricted Boltzmann machine (RBM). RBMs are most likely the most popular version of Boltzmann machine [8]. They contain one visible layer and one hidden layer. The restriction is that there is no interaction between the units of the same layer and the connections are solely between units from different layers. The Contrastive Divergence algorithm [9] has mostly been used to train the Boltzmann machine.

Big data analytics

Big Data generally refers to data that exceeds the typical storage, processing, and computing capacity of conventional databases and data analysis techniques. As a resource, Big Data requires tools and methods that can be applied to analyze and extract patterns from large-scale data. The rise of Big Data has been caused by increased data storage capabilities, increased computational processing power, and availability of increased volumes of data, which give organization more data than they have computing resources and technologies to process. In addition to the obvious great volumes of data, Big Data is also associated with other specific complexities, often referred to as the four Vs: Volume, Variety, Velocity, and Veracity. We note that the aim of this section is not to extensively cover Big Data, but present a brief overview of its key concepts and challenges while keeping in mind that the use of Deep Learning in Big Data Analytics is the focus of this paper.

The unmanageable large Volume of data poses an immediate challenge to conventional computing environments and requires scalable storage and a distributed strategy to data querying and

analysis. However, this large Volume of data is also a major positive feature of Big Data. Many companies, such as Facebook, Yahoo, Google, already have large amounts of data and have recently begun tapping into its benefits . A general theme in Big Data systems is that the raw data is increasingly diverse and complex, consisting of largely un-categorized/unsupervised data along with perhaps a small quantity of categorized/supervised data. Working with the Variety among different data representations in a given repository poses unique challenges with Big Data, which requires Big Data preprocessing of unstructured data in order to extract structured/ordered representations of the data for human and/or downstream consumption. In today's data-intensive technology era, data Velocity – the increasing rate at which data is collected and obtained – is just as important as the Volume and Variety characteristics of Big Data. While the possibility of data loss exists with streaming data if it is generally not immediately processed and analyzed, there is the option to save fast-moving data into bulk storage for batch processing at a later time. However, the practical importance of dealing with Velocity associated with Big Data is the quickness of the feedback loop, that is, process of translating data input into useable information. This is especially important in the case of time-sensitive information processing. Some companies such as Twitter, Yahoo, and IBM have developed products that address the analysis of streaming data [2]. Veracity in Big Data deals with the trustworthiness or usefulness of results obtained from data analysis, and brings to light the old adage "Garbage-In-Garbage-Out" for decision making based on Big Data Analytics. As the number of data sources and types increases, sustaining trust in Big Data Analytics presents a practical challenge.

Big Data Analytics faces a number of challenges beyond those implied by the four Vs. While not meant to be an exhaustive list, some key problem areas include: data quality and validation, data cleansing, feature engineering, high-dimensionality and data reduction, data representations and distributed data sources, data sampling, scalability of algorithms, data visualization, parallel and distributed data processing, real-time analysis and decision making, crowd sourcing and semantic input for improved data analysis, tracing and analyzing data provenance, data discovery and integration, parallel and distributed computing, exploratory data analysis and interpretation, integrating heterogenous data, and developing new models for massive data computation.

Applications of deep learning in big data analytics

As stated previously, Deep Learning algorithms extract meaningful abstract representations of the raw data through the use of an hierarchical multi-level learning approach, where in a higher-level more abstract and complex representations are learnt based on the less abstract concepts and representations in the lower level(s) of the learning hierarchy. While Deep Learning can be applied to learn from labeled data if it is available in sufficiently large amounts, it is primarily attractive for learning from large amounts of unlabeled/unsupervised data, making it attractive for extracting meaningful representations and patterns from Big Data.

Once the hierarchical data abstractions are learnt from unsupervised data with Deep Learning, more conventional discriminative models can be trained with the aid of relatively fewer supervised/labeled data points, where the labeled data is typically obtained through human/expert input. Deep Learning algorithms are shown to perform better at extracting non-local and global relationships and patterns in the data, compared to relatively shallow learning architectures [4]. Other useful characteristics of the learnt abstract representations by Deep Learning include: (1) relatively simple linear models can work effectively with the knowledge obtained from the more complex and more abstract data representations, (2) increased automation of data representation extraction from unsupervised data enables its broad application to different data types, such as image, textural, audio, etc., and (3) relational and semantic knowledge can be obtained at the higher levels of abstraction and representation of the raw data. While there are other useful aspects of Deep Learning based representations of data, the specific characteristics mentioned above are particularly important for Big Data Analytics.

Considering each of the four Vs of Big Data characteristics, i.e., Volume, Variety, Velocity, and Veracity, Deep Learning algorithms and architectures are more aptly suited to address issues related to Volume and Variety of Big Data Analytics. Deep Learning inherently exploits the availability of massive amounts of data, i.e. Volume in Big Data, where algorithms with shallow learning hierarchies fail to explore and understand the higher complexities of data patterns. Moreover, since Deep Learning deals with data abstraction and representations, it is quite likely suited for analyzing raw data presented in different formats and/or from different sources, i.e. Variety in Big Data, and may minimize need for input from human experts to extract features from every new data type observed in Big Data. While presenting different challenges for more conventional data analysis approaches, Big Data Analytics presents an important opportunity for developing novel algorithms and models to address specific issues related to Big Data. Deep Learning concepts provide one such solution venue for data analytics experts and practitioners. For example, the extracted representations by Deep Learning can be considered as a practical source of knowledge for decision-making, semantic indexing, information retrieval, and for other purposes in Big Data Analytics, and in addition, simple linear modeling techniques can be considered for Big Data Analytics when complex data is represented in higher forms of abstraction.

In the remainder of this section, we summarize some important works that have been performed in the field of Deep Learning algorithms and architectures, including semantic indexing, discriminative tasks, and data tagging. Our focus is that by presenting these works in Deep Learning, experts can observe the novel applicability of Deep Learning techniques in Big Data Analytics, particularly since some of the application domains in the works presented involve large scale data. Deep Learning algorithms are applicable to different kinds of input data; however, in this section we focus on its application on image, textual, and audio data.

Application of Big Data in Deep Learning

A key task associated with Big Data Analytics is information retrieval. Efficient storage and retrieval of information is a growing problem in Big Data, particularly since very large-scale quantities of data such as text, image, video, and audio are being collected and made available across various domains, e.g., social networks, security systems, shopping and marketing systems, defense systems, fraud detection, and cyber traffic monitoring. Previous strategies and solutions for information storage and retrieval are challenged by the massive volumes of data and different data representations, both associated with Big Data. In these systems, massive amounts of data are available that needs semantic indexing rather than being stored as data bit strings. Semantic indexing presents the data in a more efficient manner and makes it useful as a source for knowledge discovery and comprehension, for example by making search engines work more quickly and efficiently.

Instead of using raw input for data indexing, Deep Learning can be used to generate high-level abstract data representations which will

be used for semantic indexing. These representations can reveal complex associations and factors (especially when the raw input was Big Data), leading to semantic knowledge and understanding. Data representations play an important role in the indexing of data, for example by allowing data points/instances with relatively similar representations to be stored closer to one another in memory, aiding in efficient information retrieval. It should be noted, however, that the high-level abstract data representations need to be meaningful and demonstrate relational and semantic association in order to actually confer a good semantic understanding and comprehension of the input.

While Deep Learning aids in providing a semantic and relational understanding of the data, a vector representation (corresponding to the extracted representations) of data instances would provide faster searching and information retrieval. More specifically, since the learnt complex data representations contain semantic and relational information instead of just raw bit data, they can directly be used for semantic indexing when each data point (for example a given text document) is presented by a vector representation, allowing for a vector-based comparison which is more efficient than comparing instances based directly on raw data. The data instances that have similar vector representations are likely to have similar semantic meaning. Thus, using vector representations of complex high-level data abstractions for indexing the data makes semantic indexing feasible. In the remainder of this section, we focus on document indexing based on knowledge gained from Deep Learning. However, the general idea of indexing based on data representations obtained from Deep Learning can be extended to other forms of data..

2. RELATIVE WORK

Although many researchers have tackled the problem of object detection, such as the works presented in problem of creating a fast and reliable object detection system persists, as found in the survey by [14]. This is due to high variation in the appearance of the objects in field settings, including colour, shape, size, texture and reflectance properties. Furthermore, in the majority of these settings, the objects are partially abstracted and subject to continually-changing illumination and shadow conditions.

Various works presented in the literature address the problem of object detection as an image segmentation problem (i.e., object vs. background). Wang et al. [11] examined the issue of apple detection for yield prediction. They developed a system that detected apples based on their colour and distinctive specular reflection pattern. Further information, such as the average size of apples, was used to either remove erroneous detections or to split regions that could contain multiple apples. Another heuristic employed was to accept as detections only those regions that were mostly round. Bac et al. [12] proposed a segmentation approach for sweet peppers. They used a six band multi-spectral camera and used a range of features, including the raw multispectral data, normalised difference indices, as well as entropy-based texture features. Experiments in a highly controlled glasshouse environment showed that this approach produced reasonably accurate segmentation results. However, the authors noted that it was not accurate enough to build a reliable obstacle map.

Hung et al. [13] proposed the use of conditional random fields for almond segmentation. They proposed a five-class segmentation approach, which learned features using a Sparse Autoencoder (SAE). These features were then used within a CRF framework and was shown to outperform previous work. They achieved impressive segmentation performance, but did not perform object detection. Furthermore, they noted that occlusion presented a major challenge. Intuitively, such an approach is only able to cope with low levels of occlusion.

More recently, Yamamoto et al. [10] performed tomato detection by first performing colour-based segmentation. Then, colour and shape features were used to train a Classifier and Regression Trees (CART) classifier. This produced a segmentation map and grouped connected pixels into regions. Each region was declared to be a detection and to reduce the number of false alarms. They trained a non-object classifier using a random forest in controlled glasshouse environments.

In all of the above-mentioned works, a pixel-level segmentation approach for object detection has been adopted, and most of these works have examined object detection predominantly for yield estimation [8,11]. The limited studies that have conducted accurate object detection have done so for objects in controlled glasshouse environments. As such, the issue of object detection in highly challenging conditions remains unsolved. This is due to the high variability in the appearance of the target objects in the agricultural settings, which meant that the classic methods of sliding window approaches, although showing good performance when tested on datasets of selected images [15], cannot handle the variability in scale and appearance of the target objects when deployed in real farm settings.

Recently, deep neural networks have made considerable progress in object classification and detection [5, 6, 7]. The state-of-the-art detection framework on COCO [8] consists of two stages. The first stage of the pipeline applies a region proposal method, such as selective search [9] and edgebox [2] to extract regions of interest from an image and then feed them to a deep neural network for classification. Although it has high recall performance, this pipeline is computationally expensive, which prevents it from being used in real time for a robotic application. Region Proposal Networks (RPNs) [2,3] solve this problem by combining a classification deep convolutional network with the object proposal network, so the system can simultaneously predict object bounds and classify them at each position, the parameters of the two networks are shared, which results in a much faster performance, making it suitable for robotic applications.

2.1 DATASET DESIGN

Following were the details of the MRCNN project challenge datasets available for training and prediction. The dataset is generated on the background of the game of

trust which is played between two people , not necessary to be a friend or known to each other, judging whether the person should be trusted or not based on the promises which he has made and actions he has took in fulfilling of those promises all as a part of a game.

Going through the challenge details found out there are two types of dataset available:

1 RAW dataset : it sizes ranges from 26 gb to 60 gb. the size is huge because it contains lots of keypoints from feature data readings during play

2 PROCESSED dataset: size is of few kb only.

The processed dataset is derived from the raw dataset and contains the data in the format suitable for statistical, mathematical, cognitive or AI analysis. It contains the data in the form features. there are approx 100 features and few hundred rows in the training dataset. The processed dataset completely resembles the raw dataset and.

3. METHODOLOGY

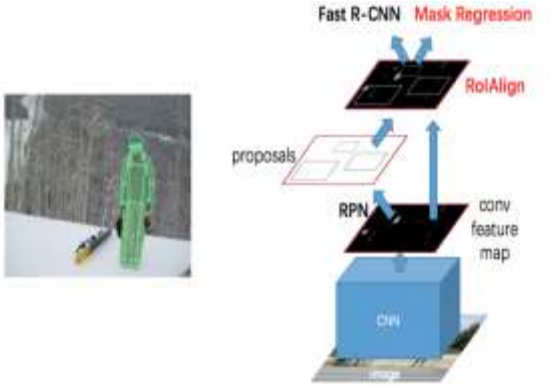

Fig 1: Diagram of Mask-RCNN

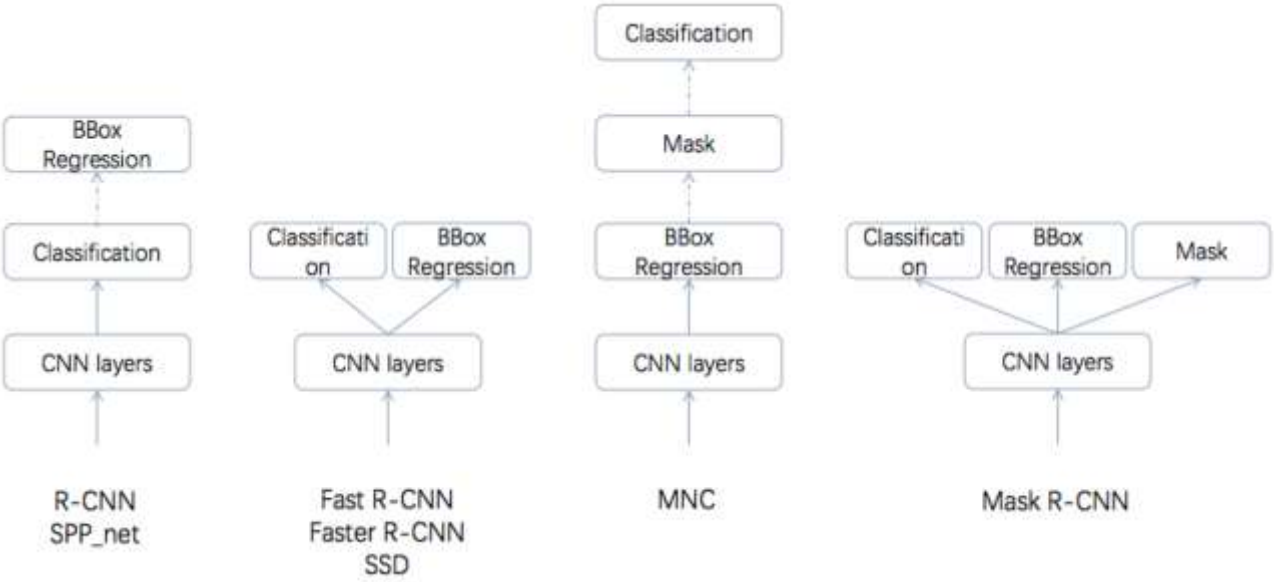

Fig 2: Diagram of various types of RCNN architectures

4. Mask RCNN

Mask R-CNN is conceptually simple: Faster R-CNN has two outputs for each candidate object, a class label and a bounding-box offset; to this we add a third branch that out-puts the object mask. Mask R-CNN is thus a natural and intuitive idea. But the additional mask output is distinct from the class and box outputs, requiring extraction of much finer spatial layout of an object. Next, we introduce the key elements of Mask R-CNN, including pixel-to-pixel alignment,

which is the main missing piece of Fast/Faster R-CNN.

Faster R-CNN:

We begin by briefly reviewing the Faster R-CNN detector [34]. Faster R-CNN consists of two stages. The first stage, called a Region Proposal Network (RPN), proposes candidate object bounding boxes. The second stage, which is in essence Fast R-CNN [12], extracts features using RoIPool from each candidate box and performs classification and bounding-box regression. The features used by both stages can be shared for faster inference. We refer readers to [21] for latest, comprehensive comparisons between Faster R-CNN and other frameworks.

5. EXPERIMENTS AND RESULTS

Various works presented in the literature address the problem of object detection as an image segmentation problem (i.e., object vs. background). Wang et al. [11] examined the issue of apple detection for yield prediction. They developed a system that detected apples based on their color and distinctive specular reflection pattern. Further information, such as the average size of apples, was used to either remove erroneous detections or to split regions that could contain multiple apples. Another heuristic employed was to accept as detections only those regions that were mostly [12] proposed a segmentation approach for sweet peppers. They used a six band multi-spectral camera and used a range of features, including the raw multispectral data, normalised difference indices, as well as entropy-based texture features. Experiments in a highly controlled glasshouse environment showed that this approach produced reasonably accurate segmentation results. However, the authors noted that it was not accurate enough to build a reliable obstacle map.

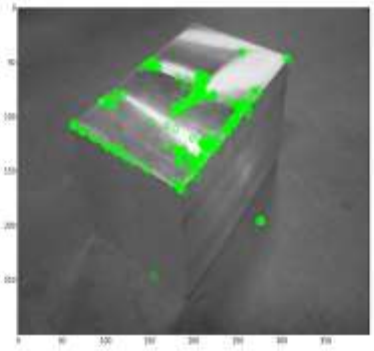

Fig 3: Feature points using Mask RCNN algorithm

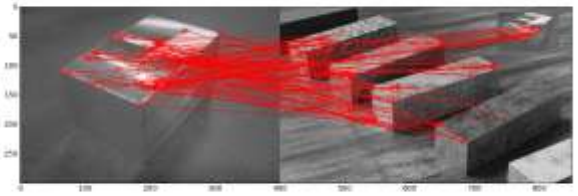

Fig 4: Diagram showing matching points of an identified object using Mask RCNN algorithm

Fig 3: Solid block input images

	Training data	AP[Val]	AP	AP50	Concrete block	Steal block	Marbel block	Wooden block
Mask-RCNN	Fine	31.5	26.2	49.9	11.7	32.9	18.7	8.4
Mask-RCNN	Fine COCO	36.4	32.0	581.	14.6	30.21	16.65	9.3

Table 1: Comparison of dataset from COCO for mask RCNN algorithm for various solid blocks

Mask R-CNN outperforms all existing, single-model entries on every task, including the COCO 2016 challenge winners. We hope our simple and effective approach will serve as a solid baseline and help ease future research in instance-level recognition. Code will be made available.

6. CONCLUSIONS

Although many researchers have tackled the problem of object detection, such as the works presented in [1.3.4], the problem of creating a fast and reliable object detection system persists, as found in the survey by [4]. This is due to high variation in the appearance of the objects in field settings, including color, shape, size, texture and reflectance properties. Furthermore, in the majority of these settings, the objects are partially abstracted and subject to continually-changing illumination and shadow conditions.

Mask R-CNN is simple to train and adds only a small overhead to Faster R-CNN, running at 5 fps. Moreover, Mask

R-CNN is easy to generalize to other tasks, e.g., allowing us to estimate human poses in the same framework. We show top results in all three tracks of the COCO suite of challenges, including instance segmentation, bounding-box object detection, and person keypoint detection. Without tricks,

Hung et al. [13] proposed the use of conditional random fields for almond segmentation. They proposed a five-class segmentation approach, which learned features using a Sparse Autoencoder (SAE). These features were then used within a CRF framework and were shown to outperform previous work. They achieved impressive segmentation performance, but did not perform object detection. Furthermore, they noted that occlusion presented a major challenge. Intuitively, such an approach is only able to cope with low levels of occlusion. More recently, Yamamoto et al. [10] performed tomato detection by first performing color-based segmentation. Then, color and shape features were used to train a MRCNN classifier. This produced a segmentation map and grouped connected pixels into regions. Each region was declared to be detection and to reduce the number of false alarms. They trained a non-object classifier using a random forest in controlled glasshouse environments.

In all of the above-mentioned works, a pixel-level segmentation approach for object detection has been adopted, and most of these works have examined object detection predominantly for yield estimation [8, 11]. The limited studies that have conducted accurate object detection have done so for objects in controlled glasshouse environments. As such, the issue of object

detection in highly challenging conditions remains unsolved. This is due to the high variability in the appearance of the target objects in the agricultural settings, which meant that the classic methods of sliding window approaches, although showing good performance when tested on datasets of selected images [15], cannot handle the variability in scale and appearance of the target objects when deployed in real farm settings.

Recently, deep neural networks have made considerable progress in object classification and detection. The state-of-the-art detection framework on COCO database consists of two stages. The first stage of the pipeline applies a region proposal method, such as selective search and edge box to extract regions of interest from an image and then feed them to a deep neural network for classification. Although it has high recall performance, this pipeline is computationally expensive, which prevents it from being used in real time for a robotic application. Region Proposal Networks (RPNs) solve this problem by combining a classification deep convolutional network with the object proposal network, so the system can simultaneously predict object bounds and classify them at each position, the parameters of the two networks are shared, which results in a much faster performance, making it suitable for robotic applications.

In real outdoor farm settings, a single sensor modality can rarely provide the needed information to detect the target objects under a wide range of variation.

7. REFERENCES

[1] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun,"Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks",IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. 39, NO. 6, JUNE 2017

[2] Cheng Wang, Ying Wang, Yinhe Han, Lili Song, Zhenyu Quan, Jiajun Li and Xiaowei Li,"CNN-based object detection solutions for embedded heterogeneous multicore SoCs", 2017 22nd Asia and South Pacific Design Automation Conference (ASP-DAC)

[3] Ross Girshick,"Fast R-CNN Object detection with Caffe",Microsoft Research

[4] Liang Zhang,Peiyi Shen 1, Guangming Zhu 1, Wei Wei 2,3 and Houbing Song,"A Fast Robot Identification and Mapping Algorithm Based on Kinect Sensor",Sensors 2015, 15, 19937-19967; doi:10.3390/s150819937

[5] Kaiming He Georgia Gkioxari Piotr Doll´ar Ross Girshick,"Mask R-CNN",Facebook AI Research (FAIR),April 2017

[6] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks",Neural Networks and signal processing ,2009 IEEE

[7] TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. 39, NO. 6, JUNE 2017

[8] .Cheng Wang, Ying Wang, Yinhe Han, Lili Song, Zhenyu Quan, Jiajun Li and Xiaowei Li,"CNN-based object detection solutions for embedded heterogeneous multicore SoCs", 2017 22nd Asia and South Pacific Design Automation Conference (ASP-DAC)

[9] Ross Girshick,"Fast R-CNNObject detection with Caffe",Microsoft Research

[10] 4.Liang Zhang,Peiyi Shen , Guangming Zhu , Wei Wei , and Houbing Song,"A Fast Robot Identification and Mapping Algorithm Based on

[11] Kinect Sensor",Sensors 2015, 15, 19937-19967; doi:10.3390/s150819937

[12] Kaiming He Georgia Gkioxari Piotr Doll´ar Ross Girshick,"Mask R-CNN",Facebook AI Research (FAIR),April 2017

[13] Maryam M Najafabadi, Flavio Villanustre, Taghi M Khoshgoftaar, Naeem Seliya, Randall WaldEmail author and Edin Muharemagic,"Deep learning applications and challenges in big data analytics",Journal of Big Data 2015

[14] Xiaojiang Peng, Cordelia Schmid,"Multi-region two-stream R-CNN for action detection",European Conference on Computer Vision, Oct 2016, Amsterdam, Netherland

[15] Sapan Naik,Bankim Patel,"Machine Vision based Fruit Classification and Grading -A Review",International Journal of Computer Applications (0975 –8887) Volume 170 –No.9, July 2017