Classification of Sugar Beet Growth Phases Using Synthetic Data in Yolo v4

Zulfiqar Ibrahim

***Abstract*—In this project, classification system has predicted different growth phases of synthetic images of sugar beet using state of an art, real-time object detection model (Yolo v4). To achieve this task, Blender is used for the creation of sugar beet's 3D models and Unreal Engine 4 for the simulation in different environments. There are four types of 3D models according to their growth phase which is then placed in a simulated environment in Unreal Engine for the acquisition of images. Unreal engine is used to simulate a farm-like environment for acquisition training images. This training data is used for the training and validation process for Yolo net. In the end, 3D models of sugar beet in a simulated environment have produced substantial results which proved that synthetic data can be used for real-life application of object detection.**

***Index Terms*—Yolo v4 (You only look once), , Unreal Engine, Blender**

I. Introduction

SUGARbeet is a very common plant that includes a high concentration of sucrose used for the production of sugar [1]. This beet belongs to the family of *Beta Vulgaris* and is normally harvested in France, Russia, Germany, and Turkey. Sugar beet produces 20% of sugar in the world and is considered a very important crop. Sugar beet consists of conical white taproot and rosette of leaves and contains 75% water and 20% and 5% pulp [2].

After the advancement in the field of A.I, private and government sectors are trying to utilize the benefits of this technology to increase the yield of the crops and preempt the spread of the diseases. Precision farming has gained substantial importance in the field of agriculture [3]. It allows an increase in the yield of crops while lowering the dependency on herbicides and pesticides. Precision farming also provides stats about corps health and identifies small portions which require treatment. Autonomous systems are being deployed in the crop field for many farming applications. These systems provide state-of-the-art services and increase crops yield.

In this project, state-of-the-art Yolo v4 net [4] was deployed for the classification of the age of the sugar beet plant in real-time. For training and validation purposes, four classes of 3D models of sugar beet were designed in Blender software and deployed in an Unreal Engine environment with variable weather settings. A camera is attached to the tractor in a simulated environment which records the video of beets planted in soil. Extracted images are labeled from the video and feed them to Yolo net for training and validation processes. The optimal goal of this project is to achieve the substantial percentage of Mean Average Precision (m.A.P) in Yolo net by using synthetic data. To achieve this goal, pipeline is divided this pipeline into IV parts. Part I is designated for the creation of 3D models of sugar beets according to their age group. In part II, these 3D models are placed into the Unreal Engine farm environment. This simulation provides two sets of videos where one video is in general RGB format and the second video is in black and red segmented format. This pixel-wise segmented format is provided by the Unreal engine. In part III, original RGB image are labeled by using a segmented image, through the OpenCV library. In part IV feed Yolo net with these labelled images for the training and validation purpose.

This paper is organized as follows: Section II explores the state of the art in relation to this project. It discusses different approaches for the classification of sugar beet's age. Section III discusses the Yolo net in detail and explores the benefits of using these algorithms over other algorithms. Section IV is about discussion of the creation of synthetic data and simulation in detail. Section V deals with the implementation of the project and training, evaluation of the Yolo net with synthetic data. Finally, section VI deals with the conclusion of the paper with a discussion about the prospects of this work.

II. Literature Review

There are many research work available for the detection of the plants according to their appearances [5]. The Deep Learning approach [6] provides more efficient and faster object detection in an uncontrolled environment which helps to extract semantic information of surroundings. Extracting semantic information about the environment is a very important topic under research in robotics. It gives the ability to make correct decisions. Other vision-based techniques are used for phenotype the plant under a very strict and controlled environment [7]. These techniques produce better results under controlled settings however observations tend to deviate from their path in an open environment.

P. Lotters et al [3]. have purposed a vision-based technique for the classification of sugar beet crops. This helps to remove herbicides and pesticides from the farm. For this task, they have employed a system that performs vegetation detection, feature extraction, random forest classification, and smoothing through a Markov random field to obtain an accurate estimate of the crops and weeds which helps the robot to value crops and the weeds must be identified by the robot's perception system to trigger the actuators for spraying or removal. Authors evaluated this approach on sugar beet plants at different growth stages and weed plants that grew on test fields near Stuttgart, Germany. As the evaluations suggest, this system provides accurate classification results. In their precision farming scenario, it is important to keep the number of false negatives, i.e., the number of sugar beet plants that are classified as weeds, small. This type of miss classification should be avoided as this would lead to the elimination of the value crop by the robot. In contrast, not detecting a weed is less critical. The evaluation of our approach suggests that the majority of weed plants get correctly classified while the number of false positives stays small.

In another paper [8], authors studied factors involved in the sub-par performance of classification systems due to the in-availability of training data. They explored solutions like transfer learning, development of temporal-spectral signatures of the target classes, and re-deployment of existing data repository or crowd-sourcing initiatives for the collection of data. The authors examined existing data entries by studying phenology information through Time-Weighted Dynamic Time Warping (TWDTW) to address the problem of automatic crop sample generation field areas. Then authors utilized proximity measures obtain through the random forest for the labels of data. High inter-class similarity tends to produce less satisfactory results. This method proved to reduce the inter-class similarity and scientists can generate crop samples with very few ground labels.

Agriculture has recently benefited from advancements in Big data, Deep learning(DL), and Robotics. These advancement has resulted in the early detection of diseases and weed plants. Many strategies have been designed and implemented due to the large acquisition and analysis of data-set involving plants. Image segmentation and machine learning approaches have been evolved to address problems attached to agriculture. In this paper [9], researchers have compared the different state of the art methodologies about plant image segmentation and machine learning at the agricultural, organ, and cellular levels in plants. Authors have shown in this paper that classification and image segmentation using these methodologies differ due to the presence of large diversity among the physical characteristics of plants. Researchers have also studied the impact of hardware technologies on these methodologies for segmentation and classification.

Muhammad Hammad Saleen et al [10]. have performed a comparative study of different ML and DL techniques used from classification in the agricultural field. Automate the feature extraction using DL methodologies has significantly improved the benefits in the agricultural field. Authors have studied the impact of DL methodologies on different agricultural applications like weed/crop discriminator, fruit counting, crop classification, disease detection. The analysis performed by the authors suggests that RCNN(Region-based Convolutional Neural Network) has produced the best plant/weed detection rate (82.25 %), compared to other famous DL algorithms like Multi-Layer Perceptron (64.9%) and K-nearest Neighbour (63.76%). Res Net-18 has performed best (94.84%) among all other DL-based. Another DL approach named FCN (Fully Convolutional Networks) produced higher accuracy (83.9%) than Support vector machines (67.6%) and Random forest (65.6%). Authors have also suggested that agricultural robots have a significant impact on DL algorithms through different data accusation techniques. Further studies need to be performed for a better understanding of the acquisition of good quality data and agricultural robots.

III. Inspection of Yolo Net

In Yolo, the Authors have developed a CNN that operates in real-time on a conventional GPU, and for which training requires only one conventional GPU. Right now there are 4 versions of Yolo exits in the market. Every preceding version performs better than the previous version. In this experimental setup, Yolo version 4 used because it performs better than 3rd version. Yolo v4 is a very efficient, powerful, and reliable real-time detector that can be trained on a single Nvidia 1080Ti or 2080Ti GPU. Authors of version 4 have used state-of-the-art methods like ”Bag of Freebies” and ”Bag of Specials” during the training process. For training purposes on a single GPU, researchers have modified existing state-of-the-art methods like CBN(Cross-iteration batch normalization) and PAN(Path aggregation network) in a very efficient way.

Object detectors developed in recent years often insert some layers between the backbone and head, and these layers are usually used to collect feature maps from different stages. Researchers call it the neck of an object detector. Usually, a neck is composed of several bottom-up paths and several top-down paths.

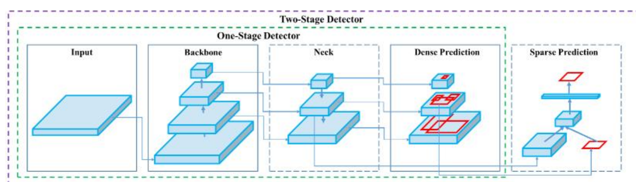

Fig. 1: Yolo Object Detection Architecture

In Yolo net, there are two types of object detection models, one stage detector and a two-stage detector. In one stage detector model, there is no need for the detection of the location of an object. one-stage detector model just predicts the bounding boxes around object wherein two-stage detector model, region of importance are detected and then classification method classify if an object in that region. Speed is an outcome from one stage detector model and accuracy ensured with two-stage detector model.

Yolo version 4 is comprised of backbone architecture and neck architecture. Backbone architecture consists of three parts, (i) Bag of Freebies (ii) Bag of specials (iii) CSP DARKNET 53.

A. *Bag of Freebies*

Bag of freebies methods [11] is those methods that change the training strategy or change the training cost to improve the model accuracy of object detection. The most important use case of these methods is to train a network in such a way that inference cost is kept as low as possible. By these methods, Volume of training data is increased while keeping the inference impact low. These methods include Data augmentation, Semantic Distribution Bias in Data-sets and Objective Function of the Bounding Box Regression.

Data augmentation is used to increase the variability of training images for a better generalization of the model. There are further methods that extend the data augmentation technique. These methods are described as Photo-metric distortion(change the variables like Brightness, Contrast, Hue, Saturation, Noise) and Geometric Distortions (change the random scaling, flipping, cropping, and rotation)

B. Bag of Specials

These are methods [12] that increase the inference cost by a small amount which increases object detection model accuracy. These methods include Mish activation, Cross Stage Partial architecture(CSP), and multi-input weighted residual connections. The bag of specials is categorized into two parts, one is Backbone and the other is Neck. Mish activation is activation function which described as f(x) = x tanh(softplus(x)). Mish activation function is bounded below and unbounded above with the range of [-0.33, +infinity]. Mish is also described as a self-regularized non-monotic activation function. These methods are considered as an add-on to further increase the accuracy of the models.

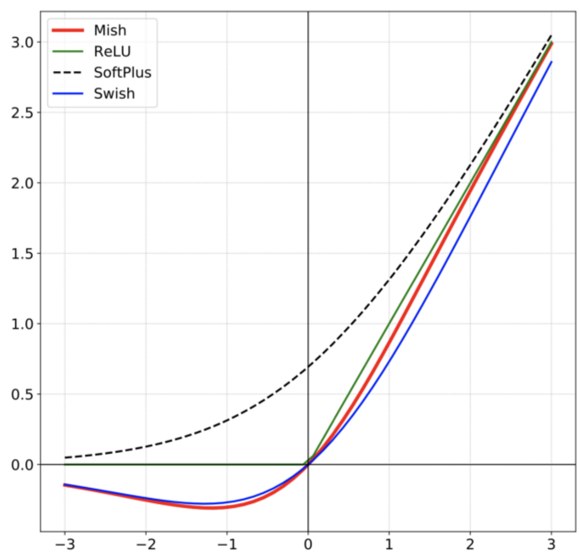

Fig. 2: Mish activation function

C. Cross Stage Partial architecture

The CSP Dark-net 53 belongs to Dense-Net architecture [13] which concatenates previous layer input to current layer input before going into the Dense layer. This solves the problem of replicated gradients which result in inefficient optimizations of a network. The problem of the replicated gradient will be resolved in the CSP dark-net by increasing the gradient flow paths which further improves the accuracy of the model. By converting the base model into two-part will result in fewer multiplications in the dense layer. This reduction in multiplication increase inference speed. These techniques allow increasing the speed of accuracy in a small inference time.

IV. Workflow

This section discusses the complete pipeline of our project. The first task in our pipeline is to create a 3D model of sugar beet. These models are categorized according to their age. there are four phases (0-4) for each 3D model. Blender software is used for the modeling of these sugar beets. For the textures of leaves, real texture is captured from a video of a sugar beet farm. In this way, there is no need to create texture from ground zero. These textures are then mapped to the leaf model of the sugar beet to create a sense of realism. After creating five models of sugar beet in Blender software, then import the .fbx files of these models in the Unreal Engine 4 for farm simulation.

In Unreal Engine 4, farm simulation is designed with variable weather. This simulation gives control over every detail like weather, soil texture, plant spacing, air pressure, terrain adjustment, etc. Tractor model is also constructed which will drive from one point to other points in the field. The path of tractor movement is selected by the user. After the path selection, camera is attached to the tractor which will take images of sugar beet planted in the soil. The parameters of a camera will be discussed later.

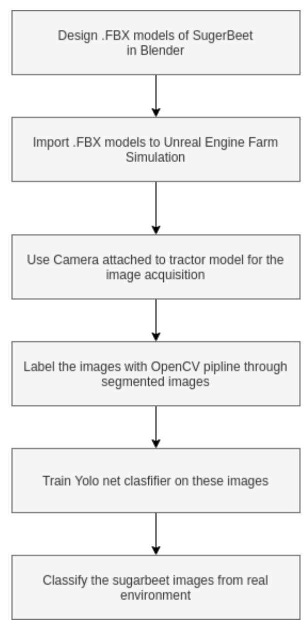

Fig. 3: Work flow of project

A. Camera Setup

The camera is the most important component in this simulation which is used for the image acquisition of sugar beet plants. The project mode of the imaging system is set as "Perspective" and Field of View is set around 90°. The reason for 90° is the capture most of the ground without the fish-eye lens effect. The aspect ratio is set around 1.777 but it can be changed according to requirements. This camera setup provides sharp images of sugar beet planted in the soil. The distance between soil and camera is around 1.5 meters.

B. 3D models of Sugar Beet

To create a simulation in Unreal Engine, a 3D model of sugar beet is designed in Blender. This is the best tool available for 3D modeling and designing. With the help of this tool, 5 phases of sugar beet are created. These 5 phases are according to their growth stages.

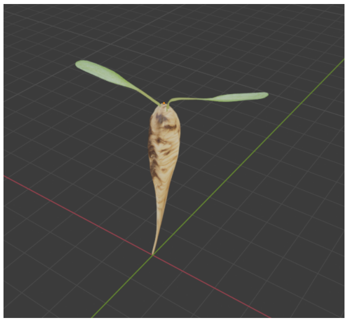

Fig. 4: Sugar Beet Phase:0

Phase 0 refers to the initial growth stage which is very early in development. This stage has very small leaves and a vertically longer beetroot.

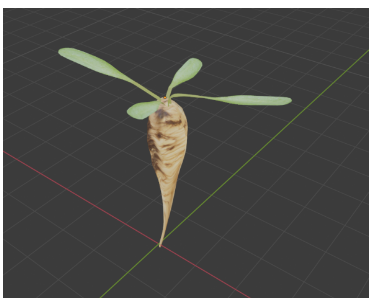

Fig. 5: Sugar Beet Phase:1

Phase 1 refers to the early stage of growth with 4 leaves. In this setting, two leaves are longer than the other two leaves. The beetroot is slightly bigger than the phase 0 beetroot.

Fig. 6: Sugar Beet Phase:2

In phase 2, mature leaves are compared to the last stage's leaves. The beetroot is more elongated as compared to the phase 1 beetroot.

The main flower is 20 % open and leaves are starting to get a yellowish-green color. All 4 leaves are properly unfolded. This growth is also named Rosette growth. Phase 3 is the stage when there are high chances of growth of infection therefore farmers use pesticides and other chemicals for the early treatment.

Fig. 7: Sugar Beet Phase:3

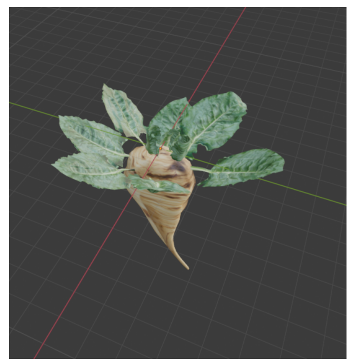

Fig. 8: Sugar Beet Phase:4

Phase 4 is the fully grown stage where the leaves are in perfect condition. Beetroot has reached harvest-able size. The main flower is fully grown and open. The main stem is approximately 20 cm long and leaves cover 90 % of the ground. After the modeling and designing of all the phases of sugar beet, object is the planted into the soil of simulated environment in Unreal Engine. This simulation provides the opportunity to create any field with any terrain of sugar beet lines. The muddy texture of soil can also be used to get the rainy environment. All of the weather conditions can be simulated using UE. This allows to create diverse data with very high variability. The size of sugar beet model can be increased/decreased to get high-quality data in real-time.

Fig. 9: Sugar beet planted in simulated farm, using UE

C. Segmentation in Unreal Engine

Segmentation of images plays a cerebral role in labeling. Yolo net requires a special format of labeling in order to train itself. The 3D models of sugar beet are being designed and planted in the simulated farm in Unreal Engine. This environment provides freedom to play around with the settings and generate any amount of data under any condition. This freedom allows to color every pixel that belongs to sugar beet's .FBX model with red and rest of the pixels will be colored black. This technique is crucial to create the segmented image of the normal RGB image. The unreal engine also provides segmented images which are very beneficial for labeling data. By using the "Sequencer" tool in Unreal Engine, cinematic is created from a farm environment. In this cinematic, images are acquired with a resolution of 416*416. Yolo net requires the resolution of training images with a multiple of 16 which is the resolution of 416*416 is selected.

Fig. 10: Segmented Vs RGB

The above segmentation technique gives the perfect opportunity to label the original with the help of the OpenCV library. Open Source Computer Vision Library (OpenCV) is an computer vision and image processing software library. OpenCV provides a common infrastructure for computer vision applications in the machine learning environment. It provides more than 2500 optimized algorithms for state-of-the-art computer vision and machine learning algorithms.

D. Labeling Pipeline

Yolo net demands special labels for training data. To achieve this task, python-based OpenCV pipeline is implemented that produces labels of each RGB image. As discussed earlier, special technique is used for the segmentation of the same data in UE. In OpenCV, same segmented image is used for the creation of the label. The format of the Yolo label can be described as a set of number which defines certain information about an object in an image. The first number suggests the class of the object and the second and third numbers describe x,y coordinates of the object in the image plane. The fourth and fifth numbers describe the height and width of the bounding box around the object.

The OpenCV pipeline ensures to draw proper bounding box region around sugar beet object in RGB image. The different morphological operations are used to remove the black spots inside the object. This allows to create a proper single blob of white pixels in a Binary image. The center(x,y) point of such blob is found using the moment's technique available in

Fig. 11: Label image with segmented image

OpenCV. Due to segmentation tool available in UE, OpenCV pipeline is able to create many labels of any object in many images. This has allowed to create a big volume of data which is helpful in the training phase.

E. Training and Validation

There are 11500 data images (divided into 5 classes) collected from the farm field environment. 75% of data is then used for training and 25% is used for validation. The Tesla P100 GPU is used with 27 GB ram in the google colab environment. This GPU has allows to train with high volume of data in a reasonable short span of time. After the training, a chart is generated which shows the graphs of Mean Average Precision(mAP) and Average Loss.

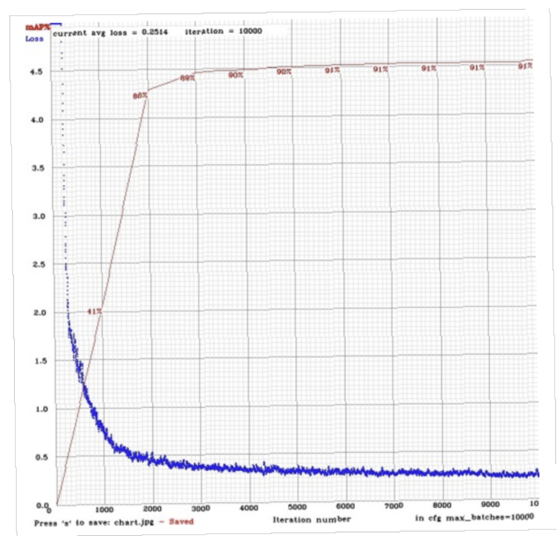

Fig. 12: Comparison chart between mAP and Average Loss

As the average loss stables, stability in mAP increases which indicates the network is converging very Finley. In the training phase, 92% of mAP is achieved which is very good for classification.

V. Results

After collecting data from simulation in an unreal engine environment, Yolo network is trained and validated. To check the classification accuracy, real-time data of sugar beet from the internet is collected and tested on Yolo network. With 92% mAP, a good score against this unseen data is achieved. If the resolution of training images are increased then there will be higher classification score.

Fig. 13: Classification score of phase 0 sugar beet

Due to the small distance between camera and object appears very large. This anomaly results in classification errors of the network.

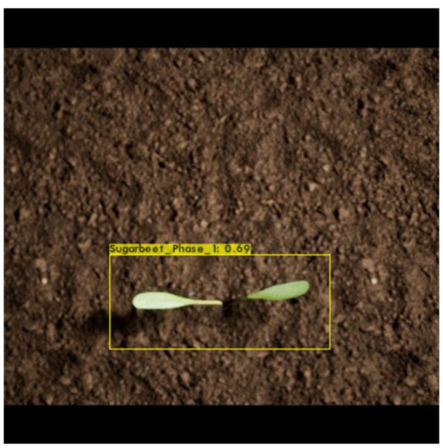

Fig. 14: Classification score of phase 1 sugar beet

This test data is from simulated farmland in an Unreal engine. As it can be seen that it has correctly classified the growth stage of the sugar beet.

Fig. 15: Classification score of phase 4 sugar beet

In Fig 15. a perfect classification score is achieved for the stage 4 sugar beet test data. This data point is from a real-time control environment with almost no sunlight. Even due to these shortcomings, the network is still able to classify the object.

VI. Future Work

Robotics and A. I have been contributing to better corps yield in agriculture. Researchers are trying to find ways for the superior use of deep learning technologies and robotics which will provide more benefits to the agricultural field.

This project is designed to use 3D models of sugar beet in a simulated farm field under variable environmental conditions. With this simulation, a high volume of synthetic data is gathered for the training and validation stage. Yolo network does not generalize well for the unseen real data as compared to unseen synthetic data in most cases. In the future, the combination of the real data with synthetic data for better classification in Yolo net is under consideration.

References

[1] A. P. Draycott, "Sugar beet," 2008.

[2] N. Islam, M. M. Rashid, S. Wibowo, C.-Y. Xu, A. Morshed, S. A. Wasimi, S. Moore, and S. M. Rahman, "Early weed detection using image processing and machine learning techniques in an australian chilli farm," *Agriculture*, vol. 11, no. 5, p. 387, 2021.

[3] P. Lottes, M. Hoeferlin, S. Sander, M. Müter, P. Schulze, and L. C. Stachniss, "An effective classification system for separating sugar beets and weeds for precision farming applications," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 2016, pp. 5157–5163.

[4] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "Yolov4: Optimal speed and accuracy of object detection," *arXiv preprint arXiv:2004.10934*, 2020.

[5] M. Aitkenhead, I. Dalgetty, C. Mullins, A. J. S. McDonald, and N. J. C. Strachan, "Weed and crop discrimination using image analysis and artificial intelligence methods," *Computers and electronics in Agriculture*, vol. 39, no. 3, pp. 157–171, 2003.

[6] A. Kamilaris and F. X. Prenafeta-Boldú, "Deep learning in agriculture: A survey," *Computers and electronics in agriculture*, vol. 147, pp. 70–90, 2018.

[7] S. Nesteruk, D. Shadrin, M. Pukalchik, A. Somov, C. Zeidler, P. Zabel, and D. Schubert, "Image compression and plants classification using machine learning in controlled-environment agriculture: Antarctic station use case," *IEEE Sensors Journal*, 2021.

[8] M. Belgiu, W. Bijker, O. Csillik, and A. Stein, "Phenology-based sample generation for supervised crop type classification," *International Journal of Applied Earth Observation and Geoinformation*, vol. 95, p. 102264, 2021.

[9] N. Zhu, X. Liu, Z. Liu, K. Hu, Y. Wang, J. Tan, M. Huang, Q. Zhu, X. Ji, Y. Jiang *et al.*, "Deep learning for smart agriculture: Concepts, tools, applications, and opportunities," *International Journal of Agricultural and Biological Engineering*, vol. 11, no. 4, pp. 32–44, 2018.

[10] M. H. Saleem, J. Potgieter, and K. M. Arif, "Automation in agriculture by machine and deep learning techniques: A review of recent developments," *Precision Agriculture*, pp. 1–39, 2021.

[11] Z. Zhang, T. He, H. Zhang, Z. Zhang, J. Xie, and M. Li, "Bag of freebies for training object detection neural networks," *arXiv preprint arXiv:1902.04103*, 2019.

[12] X. Long, K. Deng, G. Wang, Y. Zhang, Q. Dang, Y. Gao, H. Shen, J. Ren, S. Han, E. Ding *et al.*, "Pp-yolo: An effective and efficient implementation of object detector," *arXiv preprint arXiv:2007.12099*, 2020.

[13] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "Scaled-yolov4: Scaling cross stage partial network," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 13 029–13 038.