IEEE Robotics and Automation Letters (RAL) paper presented at the
2020 IEEE International Conference on Robotics and Automation (ICRA)
31 May - 31 August, 2020. Paris, France

Combining Domain Adaptation and Spatial Consistency for Unseen Fruits Counting: A Quasi-Unsupervised Approach

Enrico Bellocchio[1], Gabriele Costante[1], Silvia Cascianelli[1], Mario Luca Fravolini[1] and Paolo Valigi[1]

***Abstract*—Autonomous robotic platforms can be effectively used to perform automatic fruits yield estimation. To this aim, robots need data-driven models that process image streams and count, even approximately, the number of fruits in an orchard. However, training such models following a supervised paradigm is expensive and unpractical. Extending pre-trained models to perform yield estimation for a completely new type of fruit is even more challenging, but interesting since this situation is typical in practice. In this work, we combine a State-of-the-Art weakly-supervised fruit counting model with an unsupervised style transfer method for addressing the task above. In this sense, our proposed approach is quasi-unsupervised. In particular, we use a Cycle-Generative Adversarial Network (C-GAN) to perform unsupervised domain adaptation and train it alongside with a Presence-Absence Classifier (PAC) that discriminates images containing fruits or not. The PAC produces the weak-supervision signal for the counting network, that can then be used on the target orchard directly. Experiments on datasets collected in four different orchards show that the proposed approach is more accurate than the supervised baseline methods.**

***Index Terms*—Agricultural Automation, Robotics in Agriculture and Forestry, Visual Learning**

I. Introduction

AUTOMATION in agriculture is becoming increasingly pervasive, from the to-date commonly used automatic machinery to work in the fields and process the crop, to the more recent trend of automatizing orchard monitoring and providing support to management decisions. This last aspect can be tackled by exploiting camera-equipped robots to densely collect visual data in the orchard, and advanced Computer Vision techniques to analyse those data. Among the orchard management-related processes, yield estimation plays a crucial role in harvesting operations planning and income prevision. State-of-the-Art works [1], [2] already demonstrated how to make yield sampling cost-effective by using autonomous robots to collect images of the orchard.

This work has been supported by project "AGROBOT: robot autonomi a servizio della crescita economica e della sostenibilità ambientale dellagricoltura umbra", under call PSR 2016-2020, by Regione Umbria, submeasure 16.2.1, Focus Area 2A.

We gratefully acknowledge the support of NVIDIA Corporation with the donation of the *Titan X* GPU used for this research.

[1] Department of Engineering, University of Perugia, via Duranti 93, Perugia Italy - fax: +39 0755853654
{enrico.bellocchio, gabriele.costante, silvia.cascianelli, mario.fravolini, paolo.valigi}@unipg.it

Fig. 1: After being trained on source orchard images with weak ground truth (WS in the picture) and transformed target orchard images with no ground truth, QU-COUNT can directly count fruits from target orchard images.

However, fruit counting is still an open issue. In the case of fruits, this task can be automated via algorithms to count the fruits in the imagery collected by cameras in the fields, making it accurate and cost-effective.

Recently proposed methods [3], [4] rely on heuristics and strong supervision for fruit detection and counting. This limits the application of such methods in real agricultural contexts because of the required costs and time. Indeed, handcrafted models and heuristics are fruit specific [3], and human-generated ground truth can be unprecise [4], thus resulting in unreliable supervision information. For this reason, we follow the trend of applying the weak supervision paradigm for yield estimation [5]. This reduces the labelling work and time required to train the counting model.

Although being a more flexible approach, neither the weakly supervised paradigm guarantees that the counting model would work for fruits different from those used in training. However, the ability to generalize on different types of orchards is interesting from a commercial point of view, thus deserves more attention. Possible strategies to tackle this problem are: 1) Directly applying on target fruit $\mathcal{T}$ the model trained on source fruit $\mathcal{S}$; 2) Finetuning on target fruit $\mathcal{T}$ the model trained on source fruit $\mathcal{S}$. Both approaches have some limitations. Approach 1) would result in poor performance, thus being practically useless. Approach 2) would require strong supervision, *i.e.*, collecting and labelling image data. This is costly and time-consuming, making the approach unfeasible for agricultural applications, both because of the economic disadvantages and the risk of producing obsolete estimations.

In the sight of this, we propose to combine the recently proposed weakly supervised fruit counting model presented in [5] with an unsupervised style transfer method for addressing the situation above. In this sense, our proposed approach is quasi-unsupervised (see Fig. 1). In particular, we jointly train a Cycle-Generative Adversarial Network (C-GAN) [6] to transform images from a source orchard $\mathcal{S}$ into unpaired images of a different target orchard $\mathcal{T}$, and a Presence-Absence Classifier (PAC) to discriminate between images that contain fruits and images that do not contain fruits. The PAC is trained on real source images and synthetic source and target images from the C-GAN generators, and takes the source images ground truth as supervision signal. The trained PAC generates the weak supervision signal for the counting block, that is now able to work directly on images from the target orchard. The experimental results obtained on four different orchards demonstrate that the proposed approach gives performance comparable to a fully-supervised approach and requires shorter deploy time. For this, the approach can be considered suitable for real-world agricultural applications.

The remainder of this letter is organized as follows. In Section II, an overview of the related work is given. In Section III, the proposed approach is described. Section IV provides a detailed description of the experimental results, and conclusions are drawn in Section V.

II. Related Work

Yield estimation via fruit counting from orchard images is receiving growing interest in the Agronomics, Computer Vision and Robotics communities. Among the proposed approaches to this task, the most commonly applied are: counting by detection, counting by density estimation, and counting by regression.

The counting by detection strategy involves summing the instances detected by a class-specific object detector. This strategy is implemented *e.g.*, in [7], [8], which use low-level keypoints, in [9], [10], [11], [12], [13], which exploit segmentation, and in [14], [15], [3], [16], [17], which apply detectors based on Convolutional Neural Networks (CNN).

In the counting by density estimation strategy, object density maps are estimated along with the counting output to incorporate spatial correlation information during the training. This strategy is less applied for fruit counting. For example, in [18], the density of pixels classified as belonging to wheat spikes is estimated to infer the number of fruits.

The counting by regression strategy consists in training models to count, *i.e.*, to map image features into object instances number. This strategy was applied *e.g.*, in [19], where the number of fruits is regressed based on fruits blobs detected in the image, and in [20], where a modified version of the Inception-ResNet CNN [21] is directly trained to count.

The approaches mentioned above either rely on fruit-specific heuristics [9], [10] or on strong supervision (in the form of bound boxes [15], [3], [19], [16], [13], binary masks [11], [12], and counting labels [20], [4]), and most of them focus on a single type of fruit [15], [11], [20], [16], [13]. Recently, a weakly supervised fruit counting by regression strategy has been proposed in [5]. In this strategy, the only supervision signal needed regards simply the presence or absence of fruits in the image. The counting task is learned by combining the output of a presence-absence classifier at different locations and scales of the image, with a spatial consistency term in the training objective. This paradigm is more cost-effective than fully-supervised counterparts, and its performance is competitive.

A common situation in agricultural scenarios is the case in which yield estimation has to be performed in an unseen orchard, for a different type of fruit. In the sight of this, having an automatic yield estimation system able to deal with new domains is desirable. However, despite its practical interest, this problem has been treated only marginally in the research community. For example, in [20], synthetic images are used to train a counting network, that is then applied to real images. This approach is not flexible since it requires careful synthetic data engineering, based on heuristics on the operative conditions and the colour and shape of the fruits. In [3], transfer learning is performed by initializing the weights of a counting network for the target fruit with the weights obtained by training on the source fruit and then finetuning. However, the experimental results obtained show that this procedure has limited benefits compared to initializing the network with weights obtained by pretraining on general objects images. Different from these approaches, in this work, we treat the unseen fruit counting problem as an unsupervised domain adaptation task. In particular, we follow the State-of-the-Art adversarial domain adaptation paradigm [22], [23], [24], [25], and exploit a C-GAN [6] for generating target images. In the context of counting for agricultural applications, adversarial domain adaptation has been applied at features level in [26] for counting leaves. This approach is fully-supervised, *i.e.*, it requires precise counting ground truth and is tested in controlled environment. In contrast, the present work builds upon the counting by regression weakly-supervised method presented in [5], and extends it for the task of yield estimation in the challenging scenario of an unseen open-field orchard, for visually different types of fruits. To the best of our knowledge, this work is the first combining weak-supervision and adversarial domain adaptation for yield estimation.

III. Proposed Approach

This section describes the Quasi-Unsupervised Counting (QU-COUNT) approach. First, a general overview of the approach and the notation used are provided. Afterwards, the architecture of QU-COUNT is introduced and discussed.

A. Notation and Overview

The application context addressed by QU-COUNT comprises a source scenario $\mathcal{S}$ and a target scenario $\mathcal{T}$ that differ in the fruit species they represent. For the source scenario, we assume that a set $\mathcal{D}_{\mathcal{S}} = \{(\mathcal{I}_{\mathcal{S}}^1, c_{\mathcal{S}}^1), (\mathcal{I}_{\mathcal{S}}^2, c_{\mathcal{S}}^2), \ldots, (\mathcal{I}_{\mathcal{S}}^{Ns}, c_{\mathcal{S}}^{Ns})\}$ is available. Each image $\mathcal{I}_{\mathcal{S}}^i$ depicting a tree canopy is associated to a supervision label $c_{\mathcal{S}}^i \in \{0, 1\}$, which only indicates the presence or the absence of fruits in the image. This label is

considered *weak* with respect to the counting task since it does not express the precise number of fruit instances in the image. On the other hand, the target scenario is represented by an unsupervised set of images $\mathcal{D}_\mathcal{T} = \{\mathcal{I}_\mathcal{T}^1, \mathcal{I}_\mathcal{T}^2, \dots, \mathcal{I}_\mathcal{T}^{N_\mathcal{T}}\}$. Hence, differently from the source domain, there is no information about the presence or the absence of fruits in the images.

The driving objective of QU-COUNT is to achieve fruit counting capabilities with respect to the unlabeled target scenario $\mathcal{T}$ by using only weak knowledge about the source context $\mathcal{S}$. To fulfil this aim, it exploits two fundamental data-driven paradigms: i) *Weakly Supervised Learning* and ii) *Unsupervised Domain Adaptation*.

B. Weakly-supervised counting

Ideally, we want a network able to learn what and how to count by only relying on fruit presence-absence labels. As shown in [5], this could be achieved with a multi-branch architecture whose optimization objective is constrained by a Presence-Absence Classifier (PAC). In this work, we follow the strategy proposed in [5].

In particular, we rely on a Multi-branch Counting CNN (MBC-CNN), whose branches operate on different image tiles extracted from the original image at three different scales, *i.e.*, the full image, and the 4 and 16 non-overlapping crops. Each branch regresses the number of fruits for a given tile. During the optimization phase, the count estimates at each level are summed, and a constraint is imposed to ensure that the total count at each scale is consistent with the others. This is encoded by the following scale-consistency objective:

$$L_{SP} = \sum_{i=0}^{N} \sum_{k=0}^{2} \sum_{l=k+1}^{2} \left|\left| \left(\sum_{j=0}^{2^{2k}} \hat{y}^{(i,k,j)} \right) - \left(\sum_{m=0}^{2^{2l}} \hat{y}^{(i,l,m)} \right) \right|\right|_2^2 \quad (1)$$

where N indicates the number of training images, $\hat{y}^{(\cdot,\cdot,\cdot)}$ is the estimated fruit count for a given tile, and k and l index the different scales.

However, the loss defined by Eqn. (1) is not sufficient to avoid degenerative cases (*i.e.*, the network could simply learn to output the same value at each scale, regardless of the input image), and to ensure that the network learns to count fruits. Hence, an additional constraint is imposed by taking advantage of the presence-absence classifier. First, the PAC is separately trained by using images with the associated weak supervisory signal c^i. Afterwards, during the optimization of MBC-CNN, for each tile j at a given scale k it provides a presence-absence estimation $\hat{c}^{i,k,j}$. This information is used to impose coherence between the PAC and the MBC-CNN predictions with the following objective:

$$\begin{aligned} L_{PAC} = -\sum_{i=0}^{N} \sum_{k=0}^{2} \sum_{j=0}^{2^{2k}} & \hat{c}^{(i,k,j)} \log(g(\hat{y}^{(i,k,j)})) \\ & + (1 - \hat{c}^{(i,k,j)}) \log(1 - g(\hat{y}^{(i,k,j)})) \end{aligned} \quad (2)$$

The structures of the MBC-CNN and the PAC architectures follow for the most part the ones proposed in [5] (the reader could refer to the original paper for more details). However, in this work, the MBC-CNN is modified by introducing a Peak Stimulation Layer (PSL) inspired by [27] (see Fig. 2). The PSL is placed after the response map layer (*i.e.*, a 1×1 convolutional layer with 8 filters) on the branch that processes the full image tile. The role of this layer is to facilitate the detection of elements of interest in the image and, hence, ease the counting task. To this aim, it enhances the local maxima of the response maps and combines them to predict the presence or the absence of fruits. A binary cross-entropy (BCE) loss using the PAC prediction as the supervision signal is then computed and used to perform back-propagation, enhancing the fruit localization within the image.

C. Domain Adaptation for Fruit Counting

The strategy presented in the previous section requires presence-absence supervision labels to train the PAC. However, as introduced in Section III-A, in this work, we assume that the target scenario $\mathcal{T}$ is completely unsupervised. A naive strategy would be to use the PAC trained on $\mathcal{S}$ to constrain the optimization of the MBC-CNN on $\mathcal{T}$. Unfortunately, the two domains have different fruit species, thus, for example, a presence-absence classifier trained on olives will perform poorly on almonds or apples.

To tackle this issue, we design the QU-COUNT approach with the capability to *adapt* the PAC trained on $\mathcal{S}$ to $\mathcal{T}$ without the need for any supervision information associated with the target domain. The key intuition behind our strategy is the following: if we could translate images from $\mathcal{S}$ and change the fruit appearance and shape to resemble the species in $\mathcal{T}$, it would be possible to tune the PAC on the target domain while continuing to benefit from the presence-absence labels of the source domain.

Driven by the previous considerations, we take advantage of the recent Cycle Generative Adversarial Network (GAN) architecture [6] to achieve domain translation. The Cycle GAN (C-GAN) is then combined during the optimization phase to adapt the PAC to the target domain (see Fig. 3).

The aim of Cycle GAN is to learn two mapping functions $M : \mathcal{S} \rightarrow \mathcal{T}$ and $N : \mathcal{T} \rightarrow \mathcal{S}$ that translate images from one domain into the other. To learn these mappings, the standard GAN loss is combined with a cycle-consistency objective [6]:

$$\begin{aligned} \mathcal{L}_{GAN}(M, D_\mathcal{T}, \mathcal{S}, \mathcal{T}) = & \mathbb{E}_{I_\mathcal{T}^i \sim P_\mathcal{T}}[\log D_\mathcal{T}(I_\mathcal{T}^i)] \\ & + \mathbb{E}_{I_\mathcal{S}^i \sim P_\mathcal{S}}[\log(1 - D_\mathcal{T}(M(I_\mathcal{S}^i)))] \end{aligned} \quad (3)$$

$$\begin{aligned} \mathcal{L}_{cyc}(M, N) = & \mathbb{E}_{I_\mathcal{S}^i \sim P_\mathcal{S}}[\|N(M(I_\mathcal{S}^i)) - I_\mathcal{S}^i\|_1] \\ & + \mathbb{E}_{I_\mathcal{T}^i \sim P_\mathcal{T}}[\|M(N(I_\mathcal{T}^i)) - I_\mathcal{T}^i\|_1] \end{aligned} \quad (4)$$

where $D_\mathcal{T}$ aims to discriminate between images $I_\mathcal{T}^i \in \mathcal{T}$ and images translated from the source domain $\mathcal{S}$. Notice that a loss similar to (3) is formulated for the function $N : \mathcal{T} \rightarrow \mathcal{S}$.

However, we argue that translating images with the C-GAN framework is not sufficient to guarantee good performance on the target domain. Hence, we devise an objective function whose aim is to adapt the PAC to $\mathcal{T}$. To do so, we leverage the weakly-supervised information of source images and notice

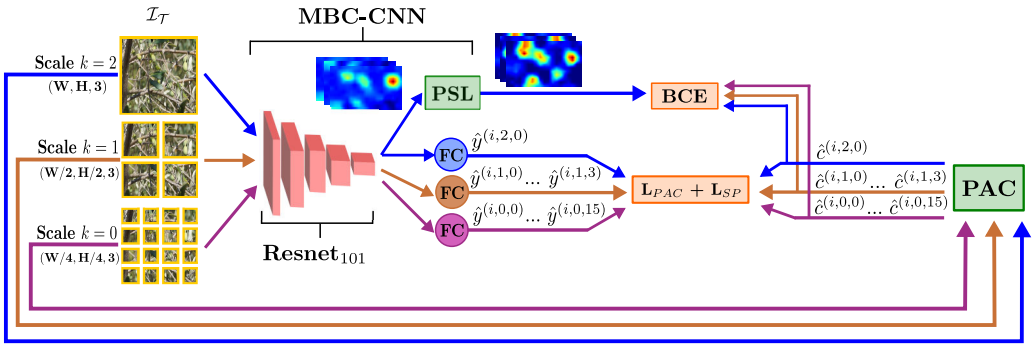


Fig. 2: Overview of the weakly supervised architecture designed to learn to count starting from weak supervision labels. Multi-branch Counting CNN processes the image at three different scales, namely the full image, and the tiles computed by considering 4 and 16 non-overlapping crops. First, the images are forwarded through a ResNet101-based feature extractor. Then, the features are fed to the peak stimulation layer (PSL) and the fully connected layers (FC). The output of the FC is the count estimate associated with each input tile, which is used to compute the losses (1) and (2).

that the information about the presence of fruits does not change when the image is translated. Thus, we can impose cross-entropy loss functions on the three images produced during the $\mathcal{S} \rightarrow \mathcal{T} \rightarrow \mathcal{S}$ cycle:

$$\begin{aligned}\mathcal{L}_{adapt} = -\frac{1}{N_s}\sum_{i=1}^{N_\mathcal{S}} \big(& c^i_\mathcal{S} \log(F(I^i_\mathcal{S})) \\ & +(1-c^i_\mathcal{S})\log(1-F(I^i_\mathcal{S})) \\ & +c^i_\mathcal{S}\log(F(I^i_{\mathcal{S}\rightarrow\mathcal{T}})) \\ & +(1-c^i_\mathcal{S})\log(1-F(I^i_{\mathcal{S}\rightarrow\mathcal{T}})) \\ & +c^i_\mathcal{S}\log(F(I^i_{\mathcal{S}\rightarrow\mathcal{T}\rightarrow\mathcal{S}}))) \\ & +(1-c^i_\mathcal{S})\log(1-F(I^i_{\mathcal{S}\rightarrow\mathcal{T}\rightarrow\mathcal{S}}))\big) \quad (5)\end{aligned}$$

The above equation combines three BCE losses that compare the ground truth presence-absence labels with the PAC predictions on source-related images, *i.e.*, $I_\mathcal{S}$, $I_{\mathcal{S}\rightarrow\mathcal{T}}$ and $I_{\mathcal{S}\rightarrow\mathcal{T}\rightarrow\mathcal{S}}$ (see Fig. 3). F represents the function learned by the PAC that maps images to presence-absence probabilities while $I^i_{\mathcal{S}\rightarrow\mathcal{T}} = M(I^i_\mathcal{S})$ and $I^i_{\mathcal{S}\rightarrow\mathcal{T}\rightarrow\mathcal{S}} = N(M(I^i_\mathcal{S}))$. Intuitively, by constraining the PAC to provide accurate prediction on the translated images $I_{\mathcal{S}\rightarrow\mathcal{T}}$, we force it to generalize also with respect to target domain images, provided that the $\mathcal{S} \rightarrow \mathcal{T}$ mapping is effectively learned.

The three losses are combined in the following objective:

$$\begin{aligned}\mathcal{L}_{tot} =& \mathcal{L}_{GAN}(M, D_\mathcal{T}, \mathcal{S}, \mathcal{T}) \\ & +\mathcal{L}_{GAN}(N, D_\mathcal{S}, \mathcal{S}, \mathcal{T}) \\ & +\mathcal{L}_{cyc}(M, N) \\ & +\mathcal{L}_{adapt} \quad (6)\end{aligned}$$

Once the PAC is adapted to the target domain, it is used to train the MBC-CNN network on $\mathcal{T}$ by following the strategy described in Section III-B.

IV. Experiments

This section describes the experiments made to validate the QU-COUNT approach. First, the datasets used and the experimental setup are illustrated. Afterwards, the results are analyzed and discussed. Finally, a possible application of the proposed counting approach in a yield estimation system is discussed and experimentally validated.

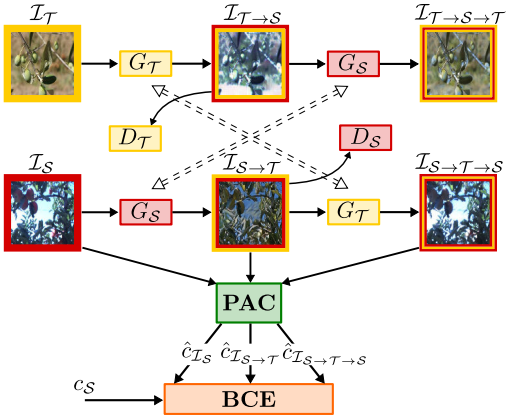


Fig. 3: Overview of the weak supervision signal generator (PAC) training. The PAC and the C-GAN are trained jointly. The C-GAN transforms the target orchard images in images that could have been taken in the source orchard and vice-versa. The PAC takes as input the source orchard images' ground truth (*i.e.*, presence-absence information), the source orchard images, and the image obtained from the C-GAN.

A. Datasets

The approach is evaluated on four different datasets containing images of three fruit species: two sets contain almond images while the other ones provide olive and apple frames.

Apples and one of the almond dataset are presented in [3]. They provide 1115 apple images at 308×202 pixels of resolution and 555 almond images at 300×300 pixels. All the samples are associated to manually labelled bounding boxes that we use to obtain the fruit count ground truth information.

The olive images and the second almond set are presented in [5]. In particular, 1158 almond images at 300×300 pixels are created by cropping 17 high-resolution frames (5472×3078 pixels) collected with a DJI Phantom drone that flies over almond trees. For olives, 1402 images with a resolution of

606×403 pixels are extracted from 31 high-quality photos taken with a professional camera. Both the sets are provided with manually annotated bounding boxes that we use to infer the counting ground truth values and the presence-absence labels.

B. Compared Approaches and Experimental Setup

To highlight the advantages introduced by QU-COUNT, we compare it against different baselines. The first one is the work introduced by [3], which proposes a deep network architecture based on FasterRCNN [28] to perform fruit detection. We also compare QU-COUNT against WS-COUNT, a state-of-the-art weakly-supervised approach presented in [5]. Both the approaches are intended to be applied in the same application domain used for training. Nevertheless, for comparison purposes, we also test them on domains that differ from the training one. Another important baseline is the MBC-CNN optimized with the PAC trained on the source domain. In this case, the presence-absence classifier is not adapted, and the images are not translated with the Cycle GAN. In the experiments, this approach is referred to as MBC-CNN+PAC$_{\mathcal{S}}$. Furthermore, to provide a reference baseline, Bargoti et al. [3], WS-COUNT [5] and MBC-CNN+PAC$_{\mathcal{S}}$ also are trained and tested in the same domain.

One could also argue whether simply translating images from $\mathcal{S}$ to $\mathcal{T}$ would be sufficient to improve the performance or not. In other words, it is important to determine whether the adaptation of the PAC adds benefits. For this reason, we consider a baseline that does not adapt the PCA, and we refer to it as MBC-CNN+GAN.

To provide an in-depth evaluation of QU-COUNT, experiments are designed to explore all possible source-target combinations of the four fruit datasets. In each test, we compute the Root Mean Square Error (RMSE) between the predicted count values and the ground truth ones. Results are statistically verified by training multiple models for each approach with different initialization seeds. The RMSEs associated with each model are used to compute means and standard deviations. p-values are also provided by using student's t-test.

C. Implementation and Training Details

The supervision information needed to train QU-COUNT and the baseline approaches is prepared by using the Pychet Labeller toolbox [3]. For the method of Bargoti et al. [3] this consists of bounding boxes for every fruit in the image, and its preparation requires on average 24 hours for a 1000 images dataset. On the other hand, the supervision signal for WS-COUNT and QU-COUNT solely consists of the indication of the presence or absence of fruits in the image, and its preparation requires less than 1 hour for a 1000 images dataset.

All the networks in the QU-COUNT framework (*i.e.*, PAC, MBC-CNN and Cycle-GAN) are implemented with the popular Pytorch framework and optimized with an Nvidia GeForce RTX 2080 Ti with 11 GB of VRAM. As detailed in Section III, the training procedure of QU-COUNT is organized into two stages. In the first one, the PAC and the Cycle-GAN are jointly optimized in order to adapt the former to the target domain. In this phase, we use the ADAM optimizer with an initial learning rate of 2×10^{-4} and a batch size of 2 samples. The Cycle GAN + PAC network is trained for 200 epochs, which takes approximately 48 hours to complete. Once the PAC is adapted to the target scenario, it is used to optimize the MBC-CNN. In this case, we train it for 40 epochs by using the Stochastic Gradient Descend (SGD) optimizer with an initial learning rate equal of 10^{-6} and setting the batch size to 4 samples. The process takes 5 hours on average. The ResNet101-based feature extraction block of the PAC and the MBC-CNN networks [29] is pretrained on ImageNet [30] and fine-tuned with a learning rate of 10^{-6}. At test time, forwarding a single image to predict the fruit count involves only the MBC-CNN network. This is the same as for the WS-COUNT approach presented in [5]. For this reason, QU-COUNT and WS-COUNT have the same computational complexity in the test phase. In particular, the average test time ranges from 0.02 seconds to 0.06 seconds per image, depending on the image resolution.

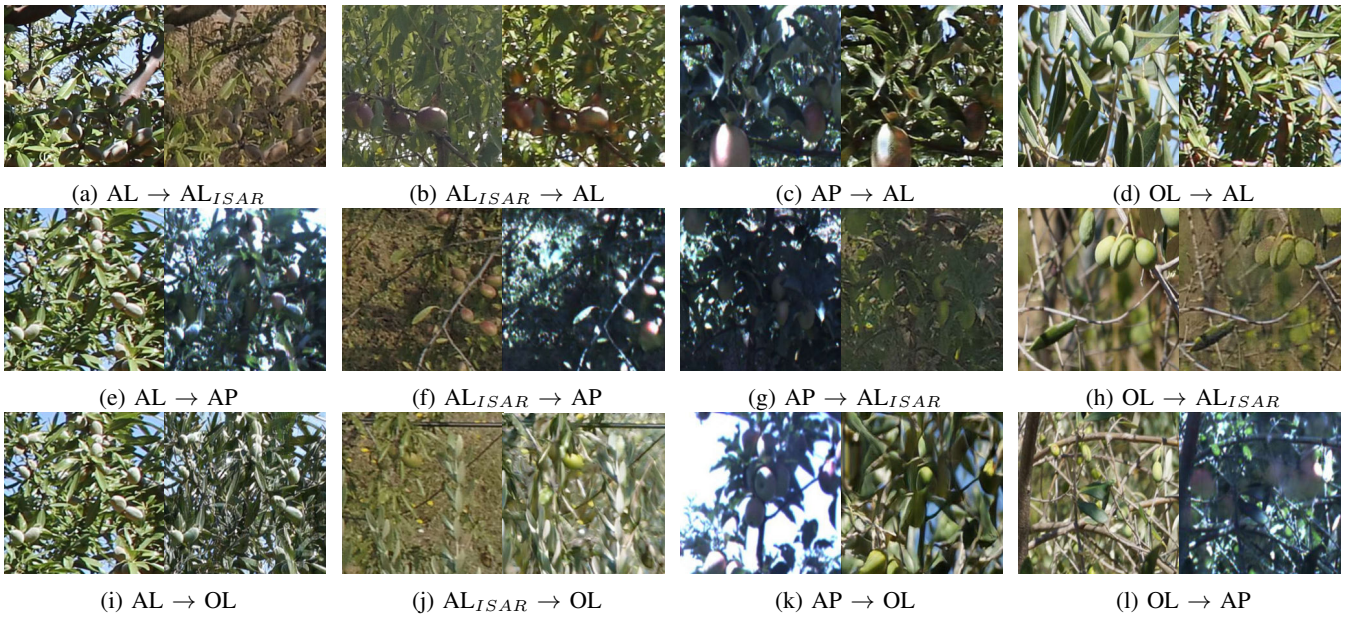
The implementation of the approach from [3] makes use of the Tensorflow implementation of Faster-RCNN. To achieve a fair comparison, the VGG16 [31] backbone is replaced with ResNet101 [29]. The detection network is initialized by using the COCO dataset [32], and different models are obtained after fine-tuning it on the fruit datasets used as source domains in the various experiments. Faster-RCNN is optimized for 100 epochs by using SGD with an initial learning rate of 3×10^{-4} and a batch size of 1 image. The training phase requires 12 hours on average , while the average test time ranges from 0.4 seconds to 0.6 seconds per image, depending on the image resolution. The implementation and the optimization of WS-COUNT follow the indications provided [5].

D. Results

Before delving into the quantitative analysis of the results, we provide a qualitative discussion on the role of the image translation step in relation to the adaptation of the PAC. Fig. 4 depicts some example of images translated across different domains. It is possible to observe that the GAN network performs two important operations: i) it adapts the image condition (*e.g.*, saturation, illumination, blur) to the target domain ones and ii) it transforms colours and shades of fruit instances to resemble the species of the target domain. Without these transformations, the PAC would not be able to "see" the properties of the target domain. Hence, it would be very challenging to detect the presence of almonds or olives if the training (*i.e.*, source) domain refers to apples (Fig. 4c and 4k).

The quantitative results are provided in TABLE I. Each one of the table blocks refers to the set of tests associated with one of the four possible target domains. Our approach assumes that no information about $\mathcal{T}$ is available, hence, we highlight in grey the performance obtained by training and testing Bargoti et al. [3], WS-COUNT [5], and MBC-CNN+PAC$_{\mathcal{S}}$ on the same domain, and use them as a reference.

The approach proposed by Bargoti et al. [3] achieves the lowest errors in all the experiments where the source and the



(a) AL $\rightarrow$ AL$_{ISAR}$ (b) AL$_{ISAR}$ $\rightarrow$ AL (c) AP $\rightarrow$ AL (d) OL $\rightarrow$ AL

(e) AL $\rightarrow$ AP (f) AL$_{ISAR}$ $\rightarrow$ AP (g) AP $\rightarrow$ AL$_{ISAR}$ (h) OL $\rightarrow$ AL$_{ISAR}$

(i) AL $\rightarrow$ OL (j) AL$_{ISAR}$ $\rightarrow$ OL (k) AP $\rightarrow$ OL (l) OL $\rightarrow$ AP

Fig. 4: Examples of image translation between source and target domains. In each group of images, the leftmost one refers to the original image from the source domain $\mathcal{I}_{\mathcal{S}}$, while the rightmost figure is its translated version with respect to the target domain, *i.e.*, $\mathcal{I}_{\mathcal{S}\rightarrow\mathcal{T}}$.

target domains are the same, with the only exception of the $OL \rightarrow OL$ test, where WS-COUNT obtains slightly better performance. These results are expected since the approach from [3] uses strong supervision signals (*i.e.*, fruit bounding boxes) during training, hence, fruit detection is more robust. On the other hand, WS-COUNT and MBC-CNN+PAC$_{\mathcal{S}}$ score similarly.

However, when a different target domain comes into play, the error of the aforementioned approaches rapidly increases in almost all cases. As an instance, consider, the $AP \rightarrow AL$ and $OL \rightarrow AL$ experiments, where the RMSE of Bargoti et al. [3], WS-COUNT [5] and MBC-CNN+PAC are fairly high, *i.e.*, the count is more than 6.00 units different from the ground truth one. Conversely, QU-COUNT provides lower errors, which proves that adapting the PAC gives considerable benefits. The same considerations apply for most of the other tests, with the exception of $AL_{ISAR} \rightarrow AL$, $AL \rightarrow AL_{ISAR}$ and $AL_{ISAR} \rightarrow AP$, where the best performance is achieved by Bargoti et al. [3]. In the first two of them, this result can be easily explained by noticing that source and target scenarios share the same fruit (*i.e.*, the almonds) and images differ only in respect to colour balance and illumination conditions (see Fig. 4a and 4b). Hence, the FasterRCNN detector trained on AL can easily manage images that come from AL_{ISAR} and vice versa.

On the other hand, the $AL_{ISAR} \rightarrow AP$ case can be understood by observing Fig. 4f. In many cases, the shape and colour of the almonds resemble those of the apple (in the dataset provided by [3], the apples have green and red shades, similar to the almonds). Thus, even if the source domain refers to almonds, it is again easy for FasterRCNN to detect the apples. Nevertheless, although in these cases the best results are obtained by [3], QU-COUNT errors are only slightly higher.

It is also important to compare the performance of MBC-CNN+GAN and QU-COUNT. Interestingly, in all the experiments QU-COUNT achieves lower error than MBC-CNN+GAN, which demonstrates that it is not sufficient to only translate images with the GAN, but it is also crucial to adapt the PAC with respect to the target domain. This is particularly true, as an instance, in the experiments where the target domain contains olives.

E. Fruit Counting for Yield Estimation

Although the focus of this work is on single image fruit counting, to draw practical considerations is interesting to show how QU-COUNT can be used to estimate the yield of an entire orchard. To this aim, we build a yield estimation system and compare the performance obtained when using either QU-COUNT or the approach from Bargoti et al. [3] as the single image counting block.

In particular, the system processes image sequences depicting rows of fruit tree and outputs the estimated yield. First, a canopy detector is run to detect trees on each frame of the image sequence. The canopy detector is based on FasterRCNN [28] with ResNet101 backbone [29], initialized by using the COCO dataset [32] and finetuned on 1000 4096×2160 pixels images collected in the same orchards considered for the datasets presented in [5]. The extracted canopy bounding boxes are then divided into 300×300 pixels tiles, zero-padded if necessary. Each tile is fed into the counting algorithm to estimate the number of fruits in the tile. The predictions on the tiles are summed to obtain the final fruit count associated with the canopy. We associate the fruit count for each tree across consecutive frames by using a Kalman-filter based tracker. The

Target domain $\mathcal{T}$: Almonds												
	AL $\rightarrow$ AL			**AL$_{ISAR}$ $\rightarrow$ AL**			**AP $\rightarrow$ AL**			**OL $\rightarrow$ AL**		
Approach	μ	σ	p-value	μ	σ	p-value	μ	σ	p-value	μ	σ	p-value
Bargoti et al. [3]	**2.19**	0.13	0.97	**3.24**	0.30	0.99	6.69	0.38	0.99	6.40	0.24	0.98
WS-COUNT [5]	4.39	1.49	0.99	4.9	1.17	0.99	6.71	0.78	0.98	6.93	0.77	0.99
MBC-CNN+PAC$_{\mathcal{S}}$	3.15	0.17	0.99	4.07	0.29	0.99	6.62	0.67	0.99	6.64	0.88	0.99
MBC-CNN+GAN	n.a.	n.a.	n.a.	6.63	0.72	0.99	4.93	0.26	0.96	5.55	0.54	0.98
QU-COUNT	n.a.	n.a.	n.a.	4.14	0.15	0.95	**3.69**	0.13	0.99	**4.20**	0.07	0.87
Target domain $\mathcal{T}$: AlmondsISAR												
	AL$_{ISAR}$ $\rightarrow$ AL$_{ISAR}$			**AL $\rightarrow$ AL$_{ISAR}$**			**AP $\rightarrow$ AL$_{ISAR}$**			**OL $\rightarrow$ AL$_{ISAR}$**		
Approach	μ	σ	p-value	μ	σ	p-value	μ	σ	p-value	μ	σ	p-value
Bargoti et al. [3]	**1.33**	0.20	0.98	**2.18**	0.20	0.98	3.11	0.11	0.99	3.45	0.14	0.98
WS-COUNT [5]	2.2	0.5	0.97	2.58	0.49	0.97	3.27	0.47	0.99	3.2	0.32	0.98
MBC-CNN+PAC$_{\mathcal{S}}$	1.96	0.10	0.85	2.43	0.14	0.99	3.45	0.35	0.96	3.62	0.28	0.99
MBC-CNN+GAN	n.a.	n.a.	n.a.	3.35	0.23	0.96	2.69	0.04	0.57	2.79	0.11	0.98
QU-COUNT	n.a.	n.a.	n.a.	2.49	0.07	0.98	**2.49**	0.03	0.78	**2.62**	0.04	0.82
Target domain $\mathcal{T}$: Apples												
	AP $\rightarrow$ AP			**AL $\rightarrow$ AP**			**AL$_{ISAR}$ $\rightarrow$ AP**			**OL $\rightarrow$ AP**		
Approach	μ	σ	p-value	μ	σ	p-value	μ	σ	p-value	μ	σ	p-value
Bargoti et al. [3]	**1.55**	0.12	0.99	3.83	0.36	0.99	**2.32**	0.08	0.97	3.72	0.29	0.98
WS-COUNT [5]	2.58	0.75	0.99	4.66	0.56	0.98	4.67	0.97	0.99	4.36	0.32	0.99
MBC-CNN+PAC$_{\mathcal{S}}$	2.36	0.05	0.69	4.98	0.48	0.99	4.75	0.65	0.99	4.82	0.18	0.95
MBC-CNN+GAN	n.a.	n.a.	n.a.	3.00	0.19	0.93	3.38	0.27	0.96	3.51	0.46	0.96
QU-COUNT	n.a.	n.a.	n.a.	**2.81**	0.10	0.94	2.56	0.07	0.98	**2.60**	0.05	0.85
Target domain $\mathcal{T}$: Olives												
	OL $\rightarrow$ OL			**AL $\rightarrow$ OL**			**AL$_{ISAR}$ $\rightarrow$ OL**			**AP $\rightarrow$ OL**		
Approach	μ	σ	p-value	μ	σ	p-value	μ	σ	p-value	μ	σ	p-value
Bargoti et al. [3]	2.32	0.18	0.99	4.16	0.27	0.97	3.35	0.38	0.99	4.34	0.40	0.99
WS-COUNT [5]	**2.21**	0.36	0.96	5.75	0.54	0.98	5.81	0.6	0.98	5.13	0.77	0.99
MBC-CNN+PAC$_{\mathcal{S}}$	2.24	0.06	0.89	5.97	0.40	0.98	5.57	0.70	0.99	4.72	0.16	0.97
MBC-CNN+GAN	n.a.	n.a.	n.a.	4.55	0.06	0.72	4.77	0.21	0.96	4.87	0.03	0.64
QU-COUNT	n.a.	n.a.	n.a.	**3.49**	0.61	0.99	**2.80**	0.08	0.94	**3.32**	0.15	0.95

TABLE I: Quantitative results achieved by the different approaches in various source-target tests. Each refers to tests on a given target domain. For each experiments, multiple models trained with different initialization seeds are used and the errors are provided as the mean RMSE (μ), its standard deviation σ and the associated p-value. The tests with same source and target domain are highlighted in gray and used as a baseline reference.

final count estimation for each tree is obtained by averaging the five highest count estimations, which reasonably have been obtained in frames where the canopy is fully captured. Finally, the total yield is simply the sum of the estimated count for each tree.

The yield estimation system is tested in two experimental sessions, in which a DJI Phantom 4 aerial drone equipped with a 4K camera captures two image streams of the same almond tree row, one from the front and the other from above. The drone flight duration is 60 seconds during which 1500 images of 4096×2160 pixels are captured. Each sequence has been manually annotated with the ground truth information about the total number of fruits. Note that the experiments are performed in the same orchard where the AlmondsISAR dataset was collected. However, the image sequences acquired for this test are not used for training QU-COUNT and Bargoti et al. [3]. To estimate the domain adaption capabilities of our approach and compare it to a baseline method, as the counting block of the yield estimation system we integrated QU-COUNT and the method of Bargoti et al. [3], both trained either on the apple or olive domains. As an additional baseline, we considered the fully supervised method of Bargoti et al. [3] trained on the same almond domain. The results of these experiments are expressed in terms of predicted count value and absolute error in comparison with the true count and reported in TABLE II. It is observed that, when the target scenario differs from the source one, QU-COUNT allows obtaining a more precise estimation of the fruit count compared to [3]. Note that the performance improvement in this experimental setup is even more evident than in the counting-only setup (see TABLE I). This suggests that QU-COUNT is robust and effective as the counting block of a complete yield estimation system. Qualitative results can be found in the video attachment and online at https://www.youtube.com/watch?v=FONROzizlZo.

V. Conclusion

In this work, we introduced QU-COUNT, a novel framework whose objective is to perform fruit counting in scenarios for which no knowledge is available by exploiting a source domain where only weak presence-absence labels are given. To this aim, we take advantage of a weakly supervised formulation and, most importantly, a domain adaptation strategy. The former ensures that the network learns to count by constraining the optimization with a Presence-Absence Classifier and multi-scale losses. The latter is responsible for the adaptation of the PAC with respect to the target domain. The benefits of QU-

Target Domain $\mathcal{T}$: AlmondsISAR					
Approach	**Source Domain $\mathcal{S}$**	**Sequence from Front**		**Sequence from Above**	
		Prediction	*Error*	*Prediction*	*Error*
QU-COUNT	**OL**	**778**	**112**	997	370
QU-COUNT	**AP**	874	208	**991**	**364**
Bargoti et al. [3]	**OL**	65	601	97	530
Bargoti et al. [3]	**AP**	41	625	108	519
Bargoti et al. [3]	**AL**$_{ISAR}$	**594**	**72**	**675**	**48**
Ground Truth		666		627	

TABLE II: Quantitative yield estimation results achieved on the two experimental sessions on the almond orchard. The results are expressed in terms of count prediction and absolute error. The tests with same source and target domain are highlighted in gray and used as a baseline reference.

COUNT are proven with experiments on four different fruits datasets, where our approach is compared against State-of-the-Art works and baselines. Furthermore, to prove that QU-COUNT can be effectively used to estimate the total yield of an orchard, we integrate it into a yield estimation system and show that it achieves top performance with respect to the State-of-the-Art baseline proposed in [3].

Future works will focus on improvements of the image translation process to ease the adaptation of the PAC and make it more robust to scenarios that considerably differ from the source one. In particular, more challenging scenarios will be addressed, such as cluster fruits and fruits with different shapes, e.g., grape and bananas. Furthermore, new loss terms based on geometrical properties and location constraints will be considered to strengthen the capability of detecting fruits and, hence, improve the counting accuracy.

REFERENCES

[1] V. Duggal, M. Sukhwani, K. Bipin, G. S. Reddy, and K. M. Krishna, "Plantation monitoring and yield estimation using autonomous quadcopter for precision agriculture," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2016, pp. 5121–5127.

[2] G. Riggio, C. Fantuzzi, and C. Secchi, "A low-cost navigation strategy for yield estimation in vineyards," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 2200–2205.

[3] S. Bargoti and J. Underwood, "Deep fruit detection in orchards," in *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 3626–3633.

[4] N. Häni, P. Roy, and V. Isler, "A comparative study of fruit detection and counting methods for yield mapping in apple orchards," *arXiv preprint arXiv:1810.09499*, 2018.

[5] E. Bellocchio, T. A. Ciarfuglia, G. Costante, and P. Valigi, "Weakly supervised fruit counting for yield estimation using spatial consistency," *IEEE Robotics and Automation Letters*, vol. 4, no. 3, pp. 2348–2355, 2019.

[6] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2242–2251.

[7] S. Nuske, K. Wilshusen, S. Achar, L. Yoder, S. Narasimhan, and S. Singh, "Automated visual yield estimation in vineyards," *Journal of Field Robotics*, vol. 31, no. 5, pp. 837–860, 2014.

[8] S. Sengupta and W. S. Lee, "Identification and determination of the number of immature green citrus fruit in a canopy under different ambient light conditions," *Biosystems Engineering*, vol. 117, pp. 51–61, 2014.

[9] K. Yamamoto, W. Guo, Y. Yoshioka, and S. Ninomiya, "On plant detection of intact tomato fruits using image analysis and machine learning methods," *Sensors*, vol. 14, no. 7, pp. 12 191–12 206, 2014.

[10] I. Sa, C. McCool, C. Lehnert, and T. Perez, "On visual detection of highly-occluded objects for harvesting automation in horticulture." ICRA, 2015.

[11] S. Bargoti and J. P. Underwood, "Image segmentation for fruit detection and yield estimation in apple orchards," *Journal of Field Robotics*, vol. 34, no. 6, pp. 1039–1060, 2017.

[12] X. Liu, S. W. Chen, S. Aditya, N. Sivakumar, S. Dcunha, C. Qu, C. J. Taylor, J. Das, and V. Kumar, "Robust fruit counting: Combining deep learning, tracking, and structure from motion," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 1045–1052.

[13] X. Liu, S. W. Chen, C. Liu, S. S. Shivakumar, J. Das, C. J. Taylor, J. Underwood, and V. Kumar, "Monocular camera based fruit counting and mapping with semantic data association," *IEEE Robotics and Automation Letters*, vol. 4, no. 3, pp. 2296–2303, 2019.

[14] I. Sa, Z. Ge, F. Dayoub, B. Upcroft, T. Perez, and C. McCool, "Deepfruits: A fruit detection system using deep neural networks," *Sensors*, vol. 16, no. 8, p. 1222, 2016.

[15] M. Stein, S. Bargoti, and J. Underwood, "Image based mango fruit detection, localisation and yield estimation using multiple view geometry," *Sensors*, vol. 16, no. 11, p. 1915, 2016.

[16] M. A. Halstead, C. S. McCool, S. Denman, T. Perez, and C. Fookes, "Fruit quantity and ripeness estimation using a robotic vision system," *IEEE Robotics and Automation Letters*, 2018.

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

[18] N. Alharbi, J. Zhou, and W. Wang, "Automatic counting of wheat spikes from wheat growth images," 2018.

[19] S. W. Chen, S. S. Shivakumar, S. Dcunha, J. Das, E. Okon, C. Qu, C. J. Taylor, and V. Kumar, "Counting apples and oranges with deep learning: A data-driven approach," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 781–788, 2017.

[20] M. Rahnemoonfar and C. Sheppard, "Deep count: fruit counting based on deep simulated learning," *Sensors*, vol. 17, no. 4, p. 905, 2017.

[21] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[22] M. Long, Z. Cao, J. Wang, and M. I. Jordan, "Conditional adversarial domain adaptation," in *Advances in Neural Information Processing Systems*, 2018, pp. 1640–1650.

[23] S. Sankaranarayanan, Y. Balaji, C. D. Castillo, and R. Chellappa, "Generate to adapt: Aligning domains using generative adversarial networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8503–8512.

[24] S. Cascianelli, G. Costante, A. Devo, T. A. Ciarfuglia, P. Valigi, and M. L. Fravolini, "The role of the input in natural language video description," *IEEE Transactions on Multimedia*, vol. 22, no. 1, pp. 271–283, Jan 2020.

[25] Q. Wang, J. Gao, and X. Li, "Weakly supervised adversarial domain adaptation for semantic segmentation in urban scenes," *IEEE Transactions on Image Processing*, 2019.

[26] M. Valerio Giuffrida, A. Dobrescu, P. Doerner, and S. A. Tsaftaris, "Leaf counting without annotations using adversarial unsupervised domain adaptation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2019, pp. 0–0.

[27] Y. Zhou, Y. Zhu, Q. Ye, Q. Qiu, and J. Jiao, "Weakly supervised instance segmentation using class peak response," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3791–3800.

[28] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, 2015, pp. 91–99.

[29] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[30] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, *et al.*, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[31] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[32] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European conference on computer vision*. Springer, 2014, pp. 740–755.