Automatic counting of cattle with Faster R-CNN on UAV images

1st João Vitor de Andrade Porto
Inovisão
Dom Bosco Catholic University
Campo Grande, Brazil
jvaporto@gmail.com

2nd Fábio Prestes Cesar Rezende
Inovisão
Dom Bosco Catholic University
Campo Grande, Brazil
fpcrezende@gmail.com

3rd Gilberto Astolfi
Inovisão
Federal Institute of Mato Grosso do Sul
Dom Bosco Catholic University
Campo Grande, Brazil
gilbertoastolfi@gmail.com

4th Vanessa Aparecida de Moraes Weber
Inovisão
State University of Mato Grosso do Sul (UEMS)
Dom Bosco Catholic University
Campo Grande, Brazil
vamoraes@gmail.com

5thMarcio Carneiro Brito Pache
Inovisão
Federal Institute of Mato Grosso do Sul
Dom Bosco Catholic University
Aquidauana, Brazil
marcio.pache@ifms.edu.br

6th Hemerson Pistori
Inovisão
Dom Bosco Catholic University
Campo Grande, Brazil
pistori@ucdb.br

***Abstract*—It is remarkable the growth of the bovine herd in the last four decades however, the availability of areas for pasture did not follow the same trend and thus caused direct interference in the binomial quality and price of the final product. One of the ways to get around this interference is by the use of technologies to help minimize the handling costs, from the breeding in a controlled environment with the need of trained manpower in the confinement process. Thus, as opposed to the current format done manually and in restricted space, computer vision technology can mitigate the identification and counting of cattle problems using unmanned aerial vehicle (UAV). Attending to the objective outlined in this article demonstrates the use of the *Faster R-CNN* for counting cattle in feedlots employing aerial images, obtaining an average precision of 89.7% for the set of hyperparameters that differed most positively from the others in this experiment.**

***Index Terms*—Computer vision, UAV, deep learning, automatic counting, Faster R-CNN.**

I. Introduction

The Brazilian beef market is one of the sectors that generate the most income for the country, currently occupying second place worldwide in meat exports. Analyzing the Brazilian bovine herd in the last four decades, there was considerable growth in the number of animals, contrasting with the pasture areas that did not keep up with the demand for meat supply and breeding space. This imbalance, even if favored by technological advances in production, has directly affected the quality of the product offered as well as increasing the price to the final consumer [1].

In Brazil, cattle counting is commonly done manually by tapering the cattle through a corral, requiring a lot of labor to minimize the occurrence of possible failures, which are becoming increasingly present with the increase in the number of cattle in increasingly smaller spaces [2].

Brazilian livestock has consolidated its position as an important producer of beef on the world stage, which is also very important for the national economic scenario, being responsible for 10% of the agribusiness gross domestic product (GDP) in 2020, and even in the face of the pandemic scenario, it has expanded its production and market coverage, resulting in an 8% increase in meat exports, especially due to the growing demand from China, in which the exported volume increased by 127% between 2019 and 2020 [3].

Considering the national economic scenario of record exports, it is evident the need for constant technological innovation in production, associated with productivity gains and economic resilience [4].

Computer vision has already been used successfully in agribusiness for classification and counting of individuals [5], [6]. So, it can help increase efficiency when applied to counting cattle, as a large number of animals could be counted automatically using an unmanned aerial vehicle (UAV).

Therefore, considering the commercial importance of beef for the national scenario, it was proposed to use UAVs to facilitate the registration of cattle in large extensions of land through aerial image captures, speeding up the process and presenting lower expense along with greater versatility than traditional methods [7].

The objective of this paper is to compare the different performances of the deep neural network *Faster R-CNN* for cattle counting using a dataset of 90 images captured by UAVs by changing its hyperparameters to define a basis for future development of a commercial tool that will perform this function.

II. Related Work

Recently computer vision methods using deep learning have emerged and obtained significant advances in the world scientific scenario. It is possible to divide such methods into two large groups: region-based methods and regression-based methods. Among the region-based methods we have the *Faster R-CNN* [8], which is able to select bounding boxes within

the image (*Bounding Boxes*) and transmit this data to a convolutional neural network (*CNN*) to classify these regions [9], [10].

Shao et al. (2019) [11], by using the neural network *YOLOv2* [12], obtained an accuracy of 95.7%, revocation of 94.6% and F-measure of 95.2% when detecting and counting bovines in the images from their two datasets one of them containing 656 images and the other with 14 both obtained in 50-meter high flights. Muribø (2019) [13], in turn, using the *YOLO* but in its third version (*YOLOv3*) obtained success in the detection of sheep in pasture using 844 thermal images (infrared) with accuracy and recall of 92% and 88% respectively.

Wang et al. (2020) [14] evaluated several detection methods for real-time monitoring of several pests attacking maize crops, using a dataset of 25.378 images of agricultural pests automatically captured by traps installed by the plantation. Among the methods chosen were YOLOv3 and *Faster R-CNN* that obtained accuracy in the act of detection of 63.54% and 51.72% respectively.

Using UAVs coupled with a convolutional neural network (*CNN*), Rivas et al. (2018) [15] obtained 95.5% accuracy in detecting cattle in the field through the use of a proprietary dataset containing 1.200 images of the animals, 1.200 images presenting their shadows and 1.200 presenting only the background. Quan et al. (2019) [16] on the other hand, achieved 97.71% accuracy when attempting to distinguish corn seedlings from noxious crop weeds through the use of *Faster R-CNN* on 62.485 images captured by their mobile robotic platform.

Xu et al. (2020) [17] applying the UAV approach, built two image datasets containing 750 specimens in each, one of cattle in the pasture and another of cattle in a feedlot, using these datasets together with the *Mask R-CNN* [18], he obtained an accuracy of 94% for the pasture dataset and 92% for the feedlot dataset.

III. Materials and Methods

A. Image Dataset

The cattle in confinement were flown over at a fixed height of 20 meters by the UAV *DJI Phantom 3 SE* capturing images automatically during the process, the flight instructions and the capture process were managed by the mobile application *Pix4D*[1]. Due to the purpose of this experiment, images that did not present bovines or images that presented only smaller parts than 10% of a bovine in its content were removed thus resulting in a total of 90 images with a dimension of 5472 x 3658 *pixels* ready for use as well as Figure 1.

These 90 images were then annotated using the *LabelImg* program through the use of *Bounding Boxes* indicating the animals in the image as shown in Figure 2, and the annotations followed the *PASCAL VOC* [19] pattern and totalled 425 different bovines.

[1]website of the tool: https://www.pix4d.com/

Fig. 1. Aerial capture of cattle in the field at a height of 20 meters.

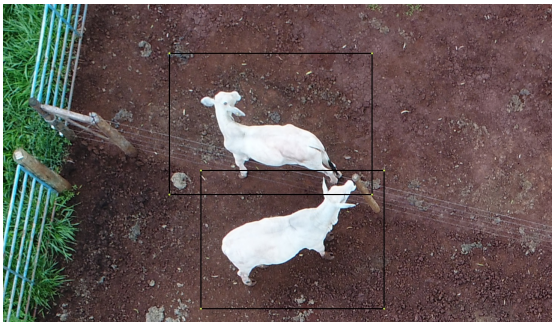

Fig. 2. Example of program *LabelImg* annotation.

B. Architecture of Faster R-CNN

The *Faster R-CNN* [8] network is a very popular architecture for multiple object detection and classification in images. This network is composed of two main modules: a region proposal network - RPN, and a *Fast R-CNN* [10] network. The *RPN* receives an image previously processed by a *CNN* to acquire an attribute map and propose possible rectangular regions in the image with three different scales and three different proportions. In this experiment, scales of 64x64, 128x128 and 256x256, and ratios of 1:1, 1:2, and 2:1 were used.

Then, the regions of interest (*ROIs*) proposed by *RPN* are read by *Fast R-CNN* and processed by a subsampling layer (*pooling layer*) resulting in attribute maps with a fixed size that is associated with a vector of attributes serving as input for a fully connected layer that has the function of classifying a given region of interest. After this classification each region of interest will have two vectors associated with it: the vector of probabilities per class, serving to indicate the class to which the region belongs, and the vector of bounding boxes, being each set of probability and bounding box corresponding to a class.

The *VGG16* is a network with 16 convolutional layers where the fixed default input of an image in RGB color space of 224x224 that passes through 5 convolution blocks whose filters are 3x3 with a fixed step of one *pixel*, each block is followed by a non-maximum reduction layer which has a 2x2 *pixel* window sliding at a step of two. In addition the network also

has a non-linear rectification layer (*ReLU*) followed by a final *Softmax* layer.

Similar to the *VGG16* architecture the *ResNet-50* also receives as input an image in RGB space with 224x224 dimensions, having a fixed convolutional filter of 3x3 with 1 *pixel* step in the convolution for almost all blocks except the first one, because it has a 7x7 filter with 2 *pixel* step. Another differential is the fact that in the *ResNet-50* occurs the process of batch normalization at the end of each convolution layer. This architecture introduces a new layer to the network called *Residual Block* that is responsible for managing the degradation process generated during training, saturating the network accuracy as the learning depth increases.

C. Experimental Design

For the experiment four experimental configurations were defined for *Faster R-CNN* using the two base networks combined to the alternation between the use and non-use of data augmentation by horizontal inversions and 90 degree rotation, generating the following approaches: *ResNet-50* without data augmentation (ResSa), *ResNet-50* with data augmentation (ResCa), *VGG16* without data augmentation (VggSa) and *VGG16* with data augmentation (VggCa). The training process of the four approaches was composed of ten epochs with an amplitude of 1000 iterations per epoch, using a step size (*stride*) equal to 16 and processing a total of 16 regions of interest simultaneously at each step. Presenting for both the classification module (*fast R-CNN*) and the region proposal module (*RPN*) a minimum overlap margin at 0.3. Two Nvidia Titan XP video cards were used for the training.

The 90 images that constitute the dataset were separated into ten groups of nine images. From this separation, the process of cross-validation of ten folds for each approach was performed, wherein each fold a different group is used as a test and the others for training and validation, following the proportion of 80% for training, 10% for validation, and 10% for a test.

After the execution of the cross-validation, the following metrics were generated: precision, revocation, accuracy, and F-measure. Thus, the four different configurations were compared to each other by means of the mean, standard deviation, one-way analysis of variance (ANOVA) with Tukey's post-test at a significance level of 5%. Furthermore, (*boxplots*) diagrams were also generated for each of the metrics, but due to the great similarity between them, only some of them were shown, as in the case of the revocation metrics, accuracy, and F-measure.

IV. Results and Discussion

From the average values and standard deviations, it was built Table I, correlating the metrics with their respective numerical values calculated and demarcating in bold the best values for each metric among the four approaches. In relation to the average performance, it can be noticed that the approaches using the *ResNet-50* obtained an advantage over the ones based on the *VGG16*, always presenting higher average values in all analyzed metrics.

Using data augmentation with *ResNet-50* as the base network caused an increase in average Precision and a decrease in the other metrics. The average Precision of ResCa at 0.897 was notably higher than the Precision of ResSa at 0.79, however, its Revocation (0.548), Accuracy (0.512), and F-measure (0.673) were lower than the values of ResSa which showed 0.759, 0.576, and 0.722 for these three metrics respectively.

Analyzing the combinations of base network and data augmentation through the metrics of Precision, Recall, Accuracy and F-measure utilizing the ANOVA test the following results were obtained: for Precision, the p-value was 0.22, in the case of Recall, the p-value resulted in 0.00134, for Accuracy a p-value equal to 0.0194 was obtained and, lastly, the F-measure obtained 0.0439 of a p-value. Since the p-value for Precision was higher than the significance level right in the ANOVA test, no post-test was performed on these data as there was no indication of any statistical difference between the four approaches, this can also be confirmed by analyzing the *boxplot* in Figure 3 that we can see the Precision of the approaches are not very far from each other, even with the median and most of the ResCa data having high values.

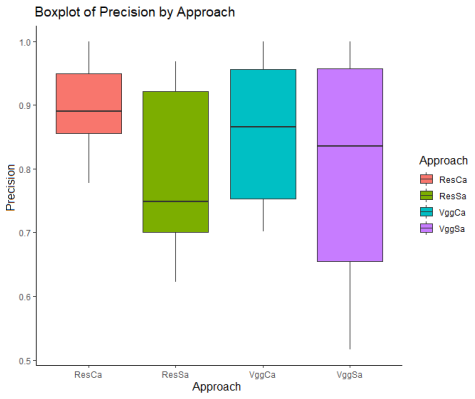

Fig. 3. *Boxplot* relating approach to Precision metric.

For presenting statistical differences among them according to their p-value, the metrics of Recall, Accuracy, and F-measure were submitted to Tukey post-test. For the case of Recall, ResSa presented a difference when compared with the other approaches, having a p-value equal to 0.04873, 0.00176 and, 0.00518 in the comparisons with ResCa, VggCa, and VggSa respectively. From the difference graph represented by Figure 4 it can be observed that the performance of ResSa was superior to the other three due to the result of the subtractions. This fact is proven when analyzing the *boxplot* present in Figure 5, which not only the median but also most of the revocation values of the ResSa approach resulted in higher than the values of the other selected combinations.

The post-test of the data related to Accuracy revealed a difference between ResSa with VggCa and VggSa with the values of 0.03861 and 0.02922 for p-value, however different from the results coming from Recall, there was no indication of difference between ResSa and ResCa as they presented a p-value equal to 0.46252 in the post-test. When observing Figure

TABLE I
RESULTS OF THE APPROACHES WITH MEAN VALUES OF EACH METRIC AND RESPECTIVE STANDARD DEVIATIONS HIGHLIGHTING IN BOLD THE BEST RESULTS

Approach	Precision	Recall	Accuracy	F-Measure
ResSa	0.79 (± 0.13)	**0.759 (± 0.08)**	**0.576 (± 0.11)**	**0.722 (± 0.09)**
VggSa	0.798 (± 0.18)	0.505 (± 0.13)	0.448 (± 0.07)	0.628 (± 0.09)
ResCa	**0.897 (± 0.07)**	0.548 (± 0.11)	0.512 (± 0.09)	0.673 (± 0.09)
VggCa	0.859 (± 0.12)	0.487 (± 0.11)	0.453 (± 0.11)	0.617 (± 0.11)

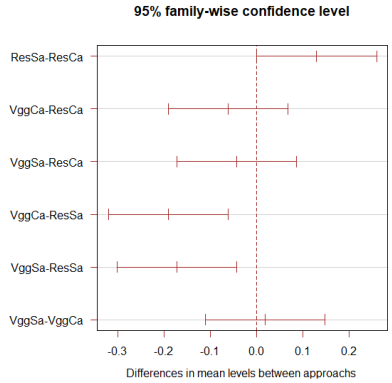

Fig. 4. Graph representing the difference in average performance levels for Recall between approaches.

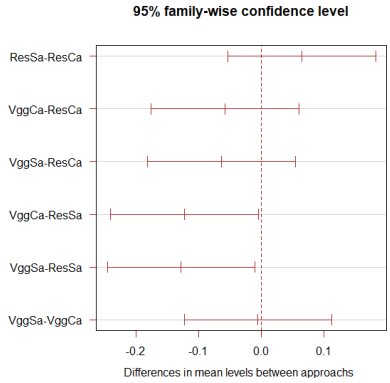

Fig. 6. Graph representing the difference in average performance levels for Accuracy between approaches

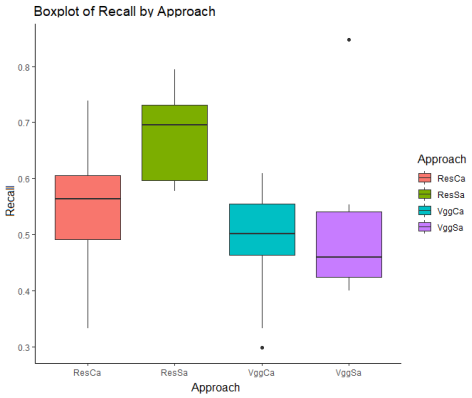

Fig. 5. *Boxplot* relating the approach to Recall metric.

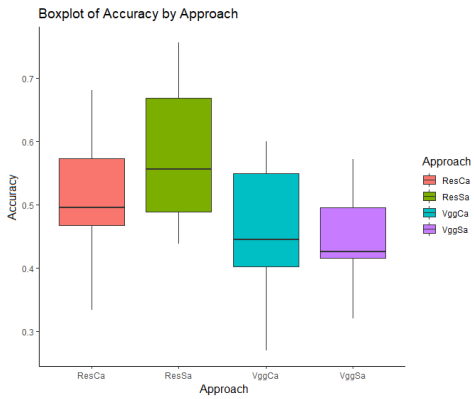

Fig. 7. *Boxplot* relating the approach to Accuracy metric.

6 it is possible to say that the performance of ResSa was superior to the performance of the two approaches using *Vgg* as a base architecture.

It is also possible to observe this indication when analyzing the *boxplot* referring to this metric represented by Figure 7 in which the values of the ResSa combination were higher than the two approaches based on the *Vgg*, but this distance is smaller when compared with ResCa.

Finally, in Figure 8 we have the Tukey results of the F-measure analysis. In this case, even with the ANOVA test presenting a significant p-value, the post-test was not able to raise evidence of a difference for the selected significance level, there were p-values close to the level as in the case of VggCa with ResSa presenting 0.05069. Even analyzing the *boxplot* related to this metric it was not possible to identify large differences between the approaches, as shown in Figure 9.

After the analysis it is possible to state that ResSa was the approach that most differed positively from the others, presenting the highest mean values of Recall, Accuracy, and F-measure. This approach has a notable statistical difference when compared to the others using the Recall and Accuracy metrics, presenting superior performance when an analysis of the difference-in-averages graphs is made. However, this approach was the one that presented the lowest average Precision among the four studied. Concerning to the F-measure, the

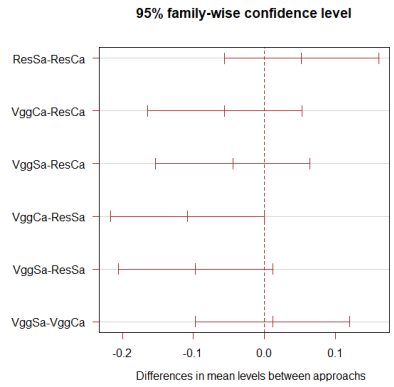

Fig. 8. Graph representing the difference in average performance levels for F-measure between approaches

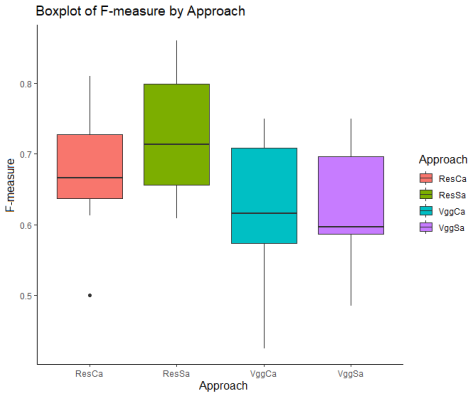

Fig. 9. *Boxplot* relating the approach to F-measure.

ANOVA test indicated statistical difference but the post-test was not able to distinguish the approaches.

Even with these differences, for this experimental configuration, the Accuracy for the four approaches could not reach large values, being always below 60%. This shows the major difference when compared to the result obtained by Rivas et al. (2018) [15] by using their *CNN* of architecture 64x64-18C7-MP4-96C5-MP2-4800L-2 followed by a multi-layer network of the type *Perceptron* (*MLP*) resulting in an Accuracy higher than 0.9. This difference is also present when comparing the results of this work with those obtained by Xu et al. (2020) [17], who also obtained Accuracies greater than 0.9 for their two image datasets (cattle in confinement and pasture) through the use of the *Mask R-CNN*.

V. Conclusion

Complex computational techniques like Deep Learning are increasingly proving to be able to perform automated tasks. So, it is possible to affirm by mean this experiment that the *Faster R-CNN* has practical applicability in the automatic counting of cattle, but the parameters defined by this work did not prove efficient enough for the formulation of the tool, mainly due to the low average accuracy among all the approaches studied. For future experiments, we propose the use of the *ResNet-50* as a base network, we propose for future analysis to fix the ResNet-50 as a backbone, changing the focus of the analysis to the variation of hyperparameters and the influence that this change will have on the learning as a whole, in addition to the increase in the database, it can also cause positive changes in the final result of a future analysis.

Acknowledgment

This work has received financial support from the Dom Bosco Catholic University the Foundation for the Support and Development of Education, Science and Technology from the State of Mato Grosso do Sul - FUNDECT (131/2016), and this study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001 and CNPq (National Council for Scientific and Technological Development) through research grants (p. 314902/2018-0). Thanks to Nvidia Corporation for donating the GPU.

References

[1] R. da Costa Gomes, G. L. D. Feijó, and L. Chiari, "Evolução e qualidade da pecuária brasileira," EMBRAPA, Tech. Rep., 03 2017.

[2] C. Bernardes, "O gado e as larguezas dos gerais," *Estudos avançados*, vol. 9, no. 23, pp. 33–58, 1995.

[3] ABIEC, "Beef report 2021," ABIEC, Tech. Rep., 06 2021. [Online]. Available: http://abiec.com.br/publicacoes/beef-report-2021/

[4] G. Bueno and M. Junior, "A sustentabilidade da pecuria brasileira." EMBRAPA, Tech. Rep., 03 2017.

[5] B. G. Weinstein, "A computer vision for animal ecology," *Journal of Animal Ecology*, vol. 87, no. 3, pp. 533–545, 2018.

[6] E. C. Tetila, B. B. Machado, N. A. de Souza Belete, D. A. Guimarães, and H. Pistori, "Identification of soybean foliar diseases using unmanned aerial vehicle images," *IEEE Geoscience and Remote Sensing Letters*, vol. 14, no. 12, pp. 2190–2194, 2017.

[7] G. H. M. Cassemiro and H. B. Pinto, "Composição e processamento de imagens aéreas de alta-resolução obtidas com drone," *Trabalho de Conclusão de Curso, Universidade de Brasília, Brasília*, 2014.

[8] 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.

[9] M. d. S. Mendes, "Aprendizado em profundidade na descrição semântica de imagens." *Trabalho de Conclusão de Curso, Universidade Federal de Ouro Preto, Ouro Preto*, 2018.

[10] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1440–1448.

[11] W. Shao, R. Kawakami, R. Yoshihashi, S. You, H. Kawase, and T. Naemura, "Cattle detection and counting in uav images based on convolutional neural networks," *International Journal of Remote Sensing*, vol. 41, no. 1, pp. 31–52, 2020.

[12] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.

[13] J. H. Muribø, "Locating sheep with yolov3," Master's thesis, NTNU, 2019.

[14] Q.-J. Wang, S.-Y. Zhang, S.-F. Dong, G.-C. Zhang, J. Yang, R. Li, and H.-Q. Wang, "Pest24: A large-scale very small object data set of agricultural pests for multi-target detection," *Computers and Electronics in Agriculture*, vol. 175, p. 105585, Aug. 2020. [Online]. Available: https://doi.org/10.1016/j.compag.2020.105585

[15] A. Rivas, P. Chamoso, A. González-Briones, and J. M. Corchado, "Detection of cattle using drones and convolutional neural networks," *Sensors*, vol. 18, no. 7, p. 2048, 2018.

[16] L. Quan, H. Feng, Y. Lv, Q. Wang, C. Zhang, J. Liu, and Z. Yuan, "Maize seedling detection under different growth stages and complex field environments based on an improved faster r–CNN," *Biosystems Engineering*, vol. 184, pp. 1–23, Aug. 2019. [Online]. Available: https://doi.org/10.1016/j.biosystemseng.2019.05.002

[17] B. Xu, W. Wang, G. Falzon, P. Kwan, L. Guo, G. Chen, A. Tait, and D. Schneider, "Automated cattle counting using mask r-cnn in quadcopter vision system," *Computers and Electronics in Agriculture*, vol. 171, p. 105300, 2020.

[18] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.

[19] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *International journal of computer vision*, vol. 88, no. 2, pp. 303–338, 2010.