Abstract. In this paper, we discuss the development of a robotic system that integrates multidisciplinary knowledge from computer vision and robotics to autonomously herd sheep. The main difficulty in sheep herding involves environmental recognition to prevent flock dispersion and to detect potential wolf appearances. Sheepdogs, as living beings, are prone to errors; therefore, the assistance of robots is necessary to enhance safety and efficiency in herding operations. The state-of-the-art YOLO model, pre-trained on the COCO dataset, has been deployed and further developed to include a new class specifically for wolves. This enhancement retains the pre-existing recognitions from the COCO dataset, while the addition of the wolf class extends the model's applicability to the livestock sector, particularly in monitoring and protecting sheep. Following this, the system is integrated into the ROS2 robotic framework, which has been one of the most widely adopted robotic systems in the last decade. ROS2's versatility allows it to support a diverse array of autonomous robots, including those operating in aerial, underwater, and subterranean environments. This work has the potential to significantly aid extensive livestock farming by streamlining daily tasks.

Keywords: livestock sector · Computer vision · Herding with robots · state-of-the-art

1 Introduction

In recent years, with the advancement of technology and rapid developments in artificial intelligence, many technologies have been applied in various fields, including agriculture [9]. Animal monitoring and management, which is the focus of this article, have particularly benefited from these advancements. For instance, drones with high-definition cameras are useful for aerial surveillance. [5], guaranteeing a rapid response to problems like wounds or the presence of raptors.

Furthermore, these technological advancements have made early disease detection and real-time health monitoring through posture and behavioral analysis possible. Nowadays, image processing is used by automated feeding systems to customize portions based on user requirements, cutting waste and maximizing growth. Additionally, breeding and intelligent monitoring are improved through the use of genetic data analysis, increasing farming practices' sustainability and efficiency

Several technologies are available to monitor sheep activities in real time [1], such as wearable GPS devices. However, these methods are costly, labor-intensive, and carry the risk of harming the sheep. Unquestionably, the use of Unmanned Aerial Vehicles (UAVs) [10] equipped with image processing technology is the most suitable method. Autonomous UAVs can efficiently collect and analyze data, detect anomalies in real time, and notify the shepherd. The data collected by these UAVs can further aid in optimizing AI models to enhance animal detection, behavioral analysis, and disease detection.

The state-of-the-art YOLO model is employed due to its speed, making it suitable for real-time tasks and surpassing previous architectures [3]. The fine-tuning technique [8] is applied to tailor the dataset specifically for the herding task.

The following sections will detail the procedures of this work, including the collection of a specialized dataset for the herding task, the retraining of the models, and the deployment to any robot equipped with ROS2-compatible hardware [6]. These steps enable a robot to recognize its environment and perform tasks autonomously.

In section 2 presents the steps involved in constructing the database.. Section 3 explains the model implementation procedure and the experiments conducted. Section 4 gathers the results obtained. And finally, Section 5 summarizes the conclusions drawn from this study.

2 Data acquisition

To build a specialized dataset for the herding task, the first step involves recording videos of four species: sheep, wolves, dogs, and people. These recordings are made from various points of view to create an enriched dataset, including close views, distant views, indoor and outdoor settings. Additionally, animals with different coat colors are included to ensure diversity. Some examples of these recordings are shown in Figure 1.

In total, we processed 38 videos of varying lengths, including 27 videos of sheep, dogs, and people, and 11 videos of wolves and people. By capturing 30 frames per second, we obtained over 80,000 images. After the segmentation annotation process, we have a total of 67,207 annotated images to build our database.

The dataset's annotation process made use of the potent 'X-Anylabeling' annotation tool, which combines state-of-the-art (SOTA) models with traditional labeling methods such as 'labelme'[7].

Currently, this vision dataset is published on the Hugging Face platform [2], a company renowned for its natural language processing (NLP) and computer vision tools and libraries. The dataset is freely available to anyone interested in utilizing it for research or development purposes.

3 Adaptation of the model for animal identification

Comparing the YOLO model to other vision models based on convolutional layers, it stands out for its high performance, adaptability, and user-friendliness, making it one of the most efficient architectures in computer vision. Object detection, classification, crawling and instance segmentation are its main tasks. We used the segmentation technique for our dataset, which provides more detailed information than just animal detection by applying a mask to each animal in addition to identifying its number and position. Very few other vision models support this feature. We used YOLOv8 and YOLOv9 in practice because the YOLOv10 segmentation version is not yet available although it was published.

Fig. 1. Samples of the images acquired from different viewpoints: close view, far view, indoor and outdoor

To utilize the YOLO model, it is common to employ transfer learning, a technique that conserves computational resources and time by using a pretrained model. YOLO is pretrained on the COCO dataset [4]. The YOLOv8 model offers five different configurations based on the number of parameters: nano (YOLOv8n), small (YOLOv8s), medium (YOLOv8m), large (YOLOv8l), and extra-large (YOLOv8x). Similarly, the YOLOv9 model provides five configurations, with only two supporting the segmentation task: large (YOLOv9c) and extra-large (YOLOv9e). As the number of model parameters increases, the performance in recognizing objects improves, but this comes at the cost of reduced training and inference speed.

In our work, we use both a small model and a large model to provide flexibility depending on the hardware capabilities of the robot. Specifically, we use the YOLOv8s model and the YOLOv9c model. We begin the training process by randomly dividing the images into 70% for training, 20% for validation, and 10% for testing. Using the recommended hyperparameters for YOLO, we employ the SGD optimizer with a learning rate of 0.01, an image input size of 640x640, and an automatic batch size (14 for YOLOv9c and 103 for YOLOv8s). Training is conducted over 50 epochs with an EarlyStopping function set to 10 epochs, which stops the training if performance does not improve for 10 consecutive epochs. The validation results are presented in Figure 1 and the test results in Figure 2.

Table 1. Results obtained the validation set by YOLOv8 and YOLOv9.

Model	Validation Set							
	Box				Mask			
	Presicion	Recall	mAP50	mAP50-95	Presicion	Recall	mAP50	mAP50-95
YOLOv8s	0.903	0.899	0.941	0.716	0.896	0.883	0.938	0.658
YOLOv9c	0.903	0.899	0.944	0.725	0.896	0.887	0.939	0.669

Table 2. Results obtained the test set by YOLOv8 and YOLOv9.

Model	Test Set							
	Box				Mask			
	Presicion	Recall	mAP50	mAP50-95	Presicion	Recall	mAP50	mAP50-95
YOLOv8s	0.903	0.913	0.947	0.724	0.901	0.88	0.935	0.638
YOLOv9c	0.904	0.913	0.949	0.732	0.899	0.889	0.936	0.654

After the first training process we observed that the model has lost all recognition of 77 other kinds of COCO, which now only recognizes sheep, people, dogs and wolves, which is not the situation we expected.

Then we have decided to freeze 20 layers of the model to just change the parameters of last model output layer, but it failed too. To address this problem we mixed 5000 COCO images to the dataset and freeze 10 able of the model, in this training process we have decided to divide the images by videos, 29 videos(55734 frames) with 4000 coco images for the train and 9 videos(11473 frames) with 1000 coco images for the validation, and the hyperparameters are kept the same as the first training. The results can be seen in Figure 3.

Table 3. Results obtained the validation set by YOLOv8 and YOLOv9 with COCO Dataset.

Model	Validation Set							
	Box				Mask			
	Presicion	Recall	mAP50	mAP50-95	Presicion	Recall	mAP50	mAP50-95
YOLOv8s	0.544	0.395	0.417	0.284	0.525	0.384	0.399	0.249
YOLOv9c	0.695	0.558	0.61	0.453	0.68	0.545	0.588	0.387
YOLOv8s-official				44.6				36.8

4 Experimental results and Discussion

By analyzing the results obtained in the previous section, we observe that the outcomes for both the validation and test data are practically identical. This indicates that the model has generalized well for this type of data. The results are very promising when using our dataset, exhibiting high accuracy in both

delineating the bounding boxes and generating the bitmap of each animal. After merge the images of the COCO dataset, the results obtained by YOLOv9 is significantly better than YOLOv9, because it has more parameters and it is an improvement of YOLOv8. Evidently the metrics worsened a lot when enlarging the dataset with the COCO images, but it is reasonable knowing that in the official documentation YOLOv8 has a mAP50-95 of boxes over 44.6% and mAP50-95 of masks over 36.8%, and our goal is to keep the recognitions of other classes and especially focused on wolves and sheep to be applied in the herding tasks, so we consider that the results obtained are considered good, the results of these two classes can be seen in the figure 4.

Table 4. Results of wolf and sheep validation sets obtained by YOLOv8 and YOLOv9 using the COCO dataset.

Model		Validation Set							
		Box				Mask			
		Presicion	Recall	mAP50	mAP50-95	Presicion	Recall	mAP50	mAP50-95
YOLOv8s	sheep	0.77	0.619	0.743	0.459	0.768	0.613	0.736	0.427
	wolf	0.853	0.67	0.798	0.569	0.855	0.669	0.799	0.548
YOLOv9c	sheep	0.821	0.695	0.816	0.548	0.816	0.687	0.808	0.51
	wolf	0.915	0.805	0.906	0.69	0.917	0.806	0.907	0.664

An experiment has also been done with the retrained models to confirm its performance to detect animals, fruits and vehicles, it has very high capacity to segment sheep, people, fruits, vehicles, and also wolves, although it can confuse many times with dogs and sheep because they have many similar characteristics, but it works sufficiently to warn shepherds when wolves approach the herd during the herding event, example of the inference of some images can be seen in figure 2.

In addition to the experiment we can look at the confusion matrix 3 to see how each class behaves individually, the resulting diagonal matrix indicates that most of the predictions of each class have been accurate.

Finally, the retrained models are deployed to a node of The Robot Operating System 2 (ROS2) using the 'YOLOv8_ros' library published on GitHub. It is capable of being used on many robots with camera sensors.

5 Conclusions

In conclusion, this work has successfully created a public dataset of sheep and wolf images, published on Hugging Face, to facilitate the automation of herding tasks using robots. We have implemented an animal recognition approach utilizing convolutional neural network models YOLOv8 and YOLOv9 to detect and segment animals in the images. This advancement enables the development of a system that can be deployed on both aerial and subterranean robots, enhancing the capabilities and applications of autonomous herding technologies.

Fig. 2. example of the use of retrained models to detect wolves, sheep, vehicles and fruits

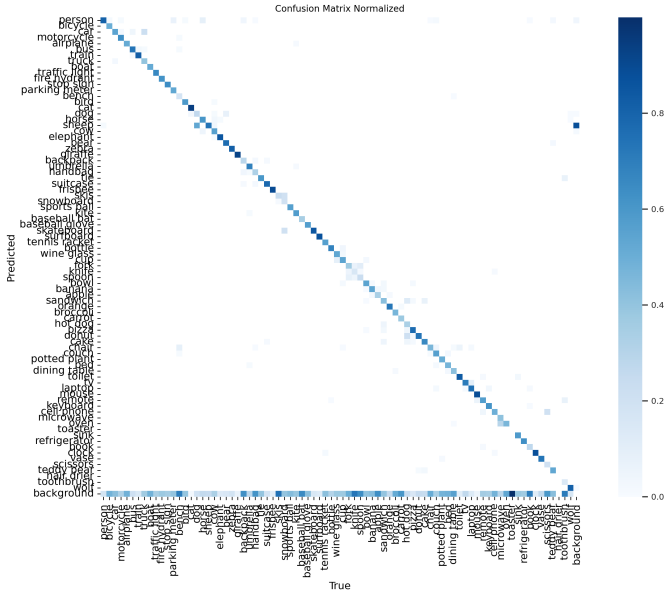

Fig. 3. normalized confusion matrix of the YOLOv9 model

The fine-tuning technique has been applied to adjust the models to the new dataset, integrating data from the COCO set and freezing half of the layers to preserve the models' previous knowledge. The results obtained demonstrate that the proposed system is effective for the detection and segmentation of sheep and wolves, significantly contributing to the automation of herding tasks and optimizing human intervention in this process

Acknowledgements

We thank all the members of the robotics team at the University of León, who provided important data and useful tools that helped make this work possible.

References

1. Carthew, S.M., Slater, E.: Monitoring animal activity with automated photography. The Journal of wildlife management pp. 689–692 (1991)
2. Jain, S.M.: Hugging face. In: Introduction to transformers for NLP: With the hugging face library and models to solve problems, pp. 51–67. Springer (2022)
3. Laroca, R., Severo, E., Zanlorensi, L.A., Oliveira, L.S., Gonçalves, G.R., Schwartz, W.R., Menotti, D.: A robust real-time automatic license plate recognition based on the yolo detector. In: 2018 international joint conference on neural networks (ijcnn). pp. 1–10. IEEE (2018)
4. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13. pp. 740–755. Springer (2014)
5. Puri, V., Nayyar, A., Raja, L.: Agriculture drones: A modern breakthrough in precision agriculture. Journal of Statistics and Management Systems **20**(4), 507–518 (2017)
6. Quigley, M., Gerkey, B., Smart, W.D.: Programming Robots with ROS: a practical introduction to the Robot Operating System. " O'Reilly Media, Inc." (2015)
7. Russell, B.C., Torralba, A., Murphy, K.P., Freeman, W.T.: Labelme: a database and web-based tool for image annotation. International journal of computer vision **77**, 157–173 (2008)
8. Tajbakhsh, N., Shin, J.Y., Gurudu, S.R., Hurst, R.T., Kendall, C.B., Gotway, M.B., Liang, J.: Convolutional neural networks for medical image analysis: Full training or fine tuning? IEEE transactions on medical imaging **35**(5), 1299–1312 (2016)
9. Talaviya, T., Shah, D., Patel, N., Yagnik, H., Shah, M.: Implementation of artificial intelligence in agriculture for optimisation of irrigation and application of pesticides and herbicides. Artificial Intelligence in Agriculture **4**, 58–73 (2020)
10. Valavanis, K.P., Vachtsevanos, G.J.: Handbook of unmanned aerial vehicles, vol. 1. Springer (2015)