Specie Identification of Common Birds in Portugal

*

João Pereira
Informatics Dept.
Faculdade de Ciências da Universidade de Lisboa
Lisbon, Portugal
joao.pereira197606@gmail.com

Duarte Gonçalves
Informatics Dept.
Faculdade de Ciências da Universidade de Lisboa
Lisbon, Portugal
duarte.dapg@gmail.com

***Abstract*—This project addresses the task of bird species classification with a focus on the common birds in Portugal. The study investigates multiple deep learning approaches, including full-image classification, segmentation-based classification, and part-based analysis using head and body crops detected via a custom-trained YOLOv8 model. Each approach leverages EfficientNet architectures trained under a consistent hyperparameter regime. While the segmented image classification treats cropped bird shapes directly, the part-based pipeline isolates anatomical regions (head and body) before classification, providing a complementary view. Evaluation is performed using macro-averaged metrics such as F1-score, AUPRC, and Top-3 Accuracy. An ensemble classifier is also implemented to explore fusion strategies across the different models. The results demonstrate the strengths and trade-offs of each method and set the foundation for robust field-ready bird recognition.**

Keywords: Bird classification, EfficientNet, YOLOv8, segmentation, part-based learning, deep learning, ensemble methods, image classification, macro-F1, species recognition

I. Problem Statement

The project aims to develop a deep learning model capable of accurately identifying images of common bird species in Portugal. This problem is particularly interesting for several reasons. Firstly, the automatic identification of bird species can help monitor bird populations and track changes in biodiversity; and, secondly, with the development of this model, there can be an integration into mobile applications to help birdwatchers identify species in real time, facilitating the job.

II. Background and Context

To provide context and background for the project, various sources will be analyzed. There are some existing applications with this type of service [1] [**?**], and it is also interesting to see how well they work for the species that will be used. To find out which species will be present in the dataset, studies have already been consulted on the most common bird species in Portugal [2]. Therefore, the species that will be present in the dataset used will be:

- *Ciconia ciconia*
- *Columbia livia*
- *Streptopelia decaocto*
- *Emberiza calandra*
- *Carduelis carduelis*
- *Serinus serinus*
- *Delichion urbicum*
- *Hirundo rustica*
- *Passer domesticus*
- *Sturnus unicolor*
- *Turdus merula*

III. Data

The project will use several existing datasets containing images of these previous species, aiming to use different numbers of images and to see what results are obtained from training, using various dataset sizes. The images will be labeled with the corresponding bird species and techniques such as rotation, inversion and cropping, applied to increase the diversity of the dataset and improve the robustness of the model.

IV. Methods and Algorithms

For the development of the project it will be implemented two parallel approaches for bird species identification. The first method employs transfer learning with pre-trained CNN architectures, EfficientNet [3], applied to whole images, enhanced with standard data augmentation techniques. The second approach will combine global image analysis with localized part-based recognition using models like YOLO models [4] for automatic detection of key anatomical features (heads, beaks, wings). Attention mechanisms will be incorporated to help the model focus on discriminative features, while standard regularization techniques will prevent overfitting.

V. Evaluation

Model evaluation in this project is centered around capturing both overall predictive performance and class-level fairness, particularly important in fine-grained classification with imbalanced data. For both the multiclass and One-vs-All binary classification tasks, the following metrics are used:

- **Macro F1-score**: Computes the F1-score independently for each class and takes the unweighted mean, ensuring that all classes contribute equally regardless of their frequency.

- **Accuracy**: Measures the proportion of correct predictions across the entire dataset, giving a general sense of the model's correctness.
- **Macro AUPRC (Area Under the Precision-Recall Curve)**: Aggregates class-wise AUPRC scores by computing the mean, giving insight into the model's precision-recall trade-off across all classes, which is particularly useful for imbalanced scenarios.
- **Top-3 Accuracy** (multiclass only): Measures how often the true class label is among the top three predicted classes with the highest probability. This metric is useful for applications where multiple suggestions can be presented to the user.
- **ROC Curve** (binary only): graphical representation of a binary classifier's performance across different classification thresholds. It plots the True Positive Rate (TPR) against the False Positive Rate (FPR), showing how the model's ability to distinguish between the two classes changes as the decision threshold varies.

To support the evaluation of the trained models beyond global metrics, both **confusion matrices** and **Grad-CAM** visualizations are used to interpret and analyze results. The confusion matrix provides a clear view of which species are most frequently confused, helping to identify specific classification weaknesses or overlaps between similar classes. Grad-CAM, on the other hand, highlights the regions in the input image that contributed most to the model's decision, offering visual insight into whether the model is focusing on meaningful features such as the bird's beak, eyes, wings, or plumage. Together, these tools provide both quantitative and qualitative understanding of model behavior.

These metrics provide a balanced view of both general and class-specific performance, enabling thorough evaluation of the models and supporting fair and informed model selection.

VI. Data Pipeline and Preprocessing

The dataset pipeline is implemented as a single automated script designed to transform raw images into a clean, standardized dataset optimized for deep learning. The process begins with data acquisition, where 600 images per species are collected from **iNaturalist** [5] and **GBIF** [6], using their respective APIs, ensuring a systematic and reproducible retrieval method.

Following data collection, the pipeline performs deduplication using perceptual hashing with a strict threshold of less than 8, effectively removing near-identical images that could bias model training. Subsequently, low-resolution images are filtered out by discarding any samples with dimensions below 200 pixels in width or height, maintaining only sufficiently detailed inputs

To address space usage, the pipeline applies data augmentation to promote a more compressed dataset for all images, this augmentation include: random horizontal flipping (p=0.5), ±20° rotation, and color jitter in HSV space. The processed images are uniformly resized to 224×224 px and stored alongside structured metadata in CSV and JSON files, while for efficient storage and fast access during training, the dataset is also compiled into an HDF5 (**Hierarchical Data Format**) file, which compresses the data while preserving its hierarchical structure.

For the segmented images model, each image is processed through a hybrid segmentation pipeline combining YOLOv8 (for bird detection) and SAM (Segment Anything Model) (for precise instance masks). Detected birds are isolated, and the masked regions are further divided vertically into three equal sections to loosely approximate anatomical regions (head, torso, tail). These image segments are then resized to 224×224 pixels and stored along with their labels in a compressed HDF5 dataset. During training, images are normalized using ImageNet statistics and augmented through random horizontal flipping, slight rotation, and color jittering. This preprocessing increases robustness to real-world variance in lighting, pose, and background. The processed tensors are organized using PyTorch, TensorDataset and loaded via DataLoader with balanced sampling.

For the part-based classification, a YOLOv8 model was trained to detect two key bird parts: the head and the body. A total of 110 images were manually annotated, with bounding boxes labeled as either bird_head or bird_body. The annotations were distributed across the 11 species in the dataset, with approximately 10 images per class, ensuring a balanced representation of bird morphology.

To train the YOLOv8 model, the dataset must follow a specific directory structure expected by the Ultralytics framework. This structure ensures that the model can correctly locate both the images and their corresponding label files during training and validation. The dataset used in this project was organized as follows:

```
datasets/annotated_dataset/
    images/
        train/
        val/
    labels/
        train/
        val/
    data.yaml
```

Each image in the images/train and images/val folders has a corresponding .txt file in the labels/train and labels/val folders, respectively. These label files follow the YOLO format: each line contains the class ID (0 for bird_head, 1 for bird_body) followed by the normalized bounding box coordinates (center_x, center_y, width, height). The data.yaml file defines the dataset configuration, including the number of classes and their names. Maintaining this structure is crucial for seamless integration with the YOLOv8 training pipeline, and ensures correct parsing of both data and annotations.

To ensure representative distribution across subsets, all datasets are partitioned, having a representation of 70% for training and 30% for validation, in the training phase and also in the results phase. It is planned that the test group will be

set up in the future, with field photographs if possible, and all the models will be evaluated in a final test.

VII. Trained Models

A. Full Images Model

For the full images model training, we performed a structured grid search comparing EfficientNet-B0 and EfficientNet-V2-S architectures. The search systematically evaluated critical hyperparameters including dropout rate, to prevent overfitting, learning rate, for optimal convergence, and weight decay, for effective regularization. After analysing the results given, the best hyperparameters are:

- **Architecture**: EfficientNet-V2-S
- **Optimizer**: Adamw
- **Learning Rate**: 1e-4
- **Weight Decay**: 1e-4
- **Dropout**: 0.0
- **Batch Size**: 32
- **Epochs**: 25

B. Segmented Images Model

The segmented model is based on EfficientNet-B0, pretrained on ImageNet. The early layers are frozen, and only the last three convolutional blocks are fine-tuned. The final classification head is replaced with a dropout layer followed by a fully connected layer with output size matching the number of classes. Analising the results from gridsearch process, this are the hyperparameters chosen:

- **Architecture**: EfficientNet-B0
- **Loss Function**: CrossEntropyLoss
- **Optimizer**: Adam
- **Learning Rate**: 1e-3
- **Weight Decay**: 1e-4
- **Dropout**: 0.5
- **Batch Size**: 32
- **Epochs**: 25

Training also follows a OneCycle learning rate policy with cosine annealing. No early stopping is used, but the best model is selected based on macro-F1 and top-3 accuracy.

C. Part-Based Model

1) YOLOv8 Part Detection (Head + Body): The model was trained using the YOLOv8n (nano) configuration, which offers a balance between speed and performance. The following hyperparameters were used:

- **Model**: yolov8n.pt (pretrained)
- Epochs: 50
- Image size: 640 × 640
- Batch size: 16
- Optimizer: auto (default)

Once trained, the YOLOv8 model was used to automatically generate head and body crops from the full bird image dataset, enabling the training of specialized part-based classifiers.

2) Head and Body Classifier Training: Two separate EfficientNet-B0 models were trained: one using only the cropped head regions and another using only the body regions.

Each model was trained as a standard multiclass classifier with 11 output classes, using the same architecture and hyperparameters as the segmented image model:

- **Architecture**: EfficientNet-B0
- **Loss Function**: CrossEntropyLoss
- **Optimizer**: Adam
- **Learning Rate**: 1e-3
- **Weight Decay**: 1e-4
- **Dropout**: 0.5
- **Batch Size**: 32
- **Epochs**: 25

This part-based setup provides complementary visual perspectives of the bird and is used later in ensemble and fusion strategies to enhance overall classification performance.

VIII. Results

A. Full Images Model

In the initial attemps, the model demonstrated strong performance in bird species classification across multiple evaluation metrics and iterations:

- Test Loss: 0.69
- Top-1 Accuracy: 0.81
- Top-3 Accuracy: 0.95
- Macro F1-score: 0.81
- Macro-AUPRC: 0.98

In terms of computational efficiency, the training process completed in approximately 25.7 minutes while utilizing 2.77 GB of memory, demonstrating that the model achieves strong predictive performance without excessive resource demands.

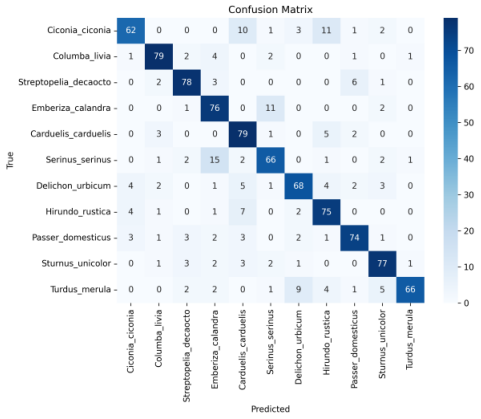

Fig. 1. Confusion Matrix, 11 classes (Full Images)

The confusion matrix (fig. 1) revealed high precision for species like *Columbia_livia* and *Carduellis_carduellis*, with 79 correct. Key misclassifications occurred between visually similar species, such as *Serinus_serinus* being confused with *Emberiza_calandra* and *Delichon_urbicum* with *Hirundo_rustica*. Beside this patterns error the core data demonstrates the

model's capability to distinguish most species while highlighting specific challenging cases that could benefit from targeted improvements in future iterations.

In the final execution, despite fluctuations during training, the best overall results were achieved:

- Test Loss: 0.86
- Top-1 Accuracy: 0.83
- Top-3 Accuracy: 0.95
- Macro F1-score: 0.83
- Macro-AUPRC: 0.98

In this confusion matrix (fig. 2),the challenges persisted including misclassifications of *Serinus_serinus* with *Emberiza_calandra* and *Carduelis_carduelis* with *Hirundo_rustica*, suggesting the need for targeted architectural adjustments or data enrichment for these species. The rest of the classes seemed to be capable to be distinguished.

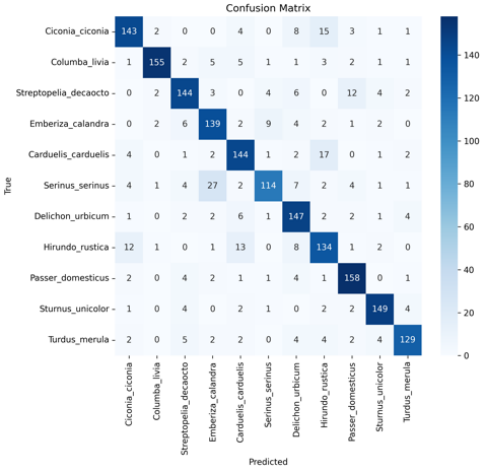

Fig. 2. Confusion Matrix, 11 classes (Full Images - Final Execution)

B. Segmented Images Model

The multiclass classification model achieved the following scores:

- Macro F1-score: ~ 0.74
- Accuracy: ~ 0.75
- Macro-AUPRC: ~ 0.83
- Top-3 Accuracy: ~ 0.91

These results indicate strong balanced performance, with top-3 accuracy showing the model reliably includes the correct label among its top suggestions. Confusion matrices (fig. 3) reveal strong precision on visually distinct species and some misclassifications between similar ones (e.g., Columba livia and Streptopelia decaocto).

One way of obtaining more reliable results is to create a 12th class, Uncertain, and classifications with a probability lower than a certain threshold are classified as being in this class. This way, when the probability of an image being of a certain class is low, it is prevented from being classified as being of that class.

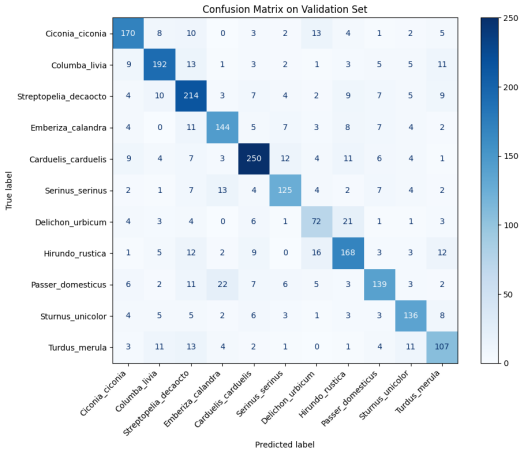

Fig. 3. Confusion matrix, 11 classes (Segmented Images)

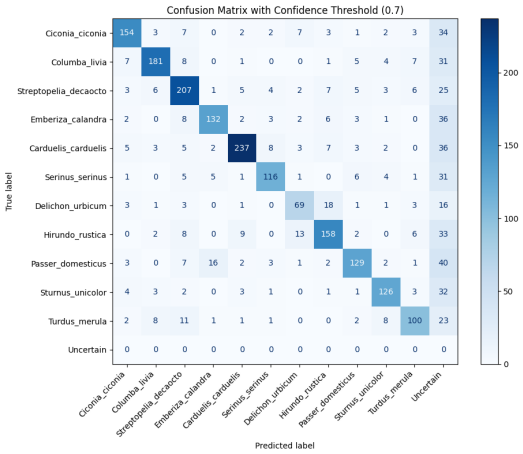

Fig. 4. Confusion matrix, 11 classes and Uncertain class

Analyzing the confusion matrix with a confidence threshold of 0.7 (fig. 4), it's possible to understand that the model remains highly accurate across most classes while now deferring a notable number of uncertain predictions. This thresholding improves prediction reliability by reducing misclassifications, especially for classes like Columba livia, *Passer domesticus*, and *Turdus merula*, which previously had higher confusion rates. Although accuracy for some classes slightly decreases, the addition of the "Uncertain" category introduces a safety margin for low-confidence predictions, enhancing the model's robustness and interpretability in practical applications.

In parallel, a One-vs-All strategy was implemented, training one binary classifier per species. These models were trained using the same architecture and configuration, with undersampling of the negative class to maintain class balance. Evaluation across the 11 binary models yielded:

- Average Macro F1-score: ~ 0.88
- Average Accuracy: ~ 0.88
- Average Macro AUPRC: ~ 0.94

C. Part Detection + Classification Model

1) YOLO model to detect bird's head and body: The results of the YOLO model trained to recognize the birds' heads and bodies can now be analyzed. In the prediction image (fig 5), the model correctly identifies the two target classes. The bounding boxes around the bird's head and body are well-localized, suggesting that the manual annotations and training setup were sufficient for teaching the model the discriminative features of each part.

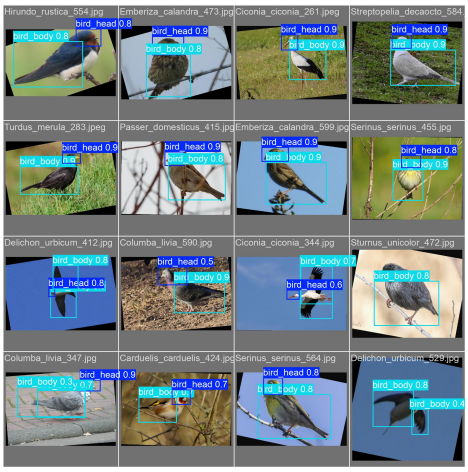

Fig. 5. Some examples of predictions, validation group

The normalized confusion matrix (fig. 6) further supports these findings. The matrix indicates high true positive rates for both classes (bird_head and bird_body), with minimal misclassifications. The clarity and separation between the two classes in the matrix imply that the model can differentiate the head from the body with a high degree of accuracy. Any off-diagonal activity is minimal, reflecting low inter-class confusion, which is particularly encouraging given the fine-grained nature of the task.

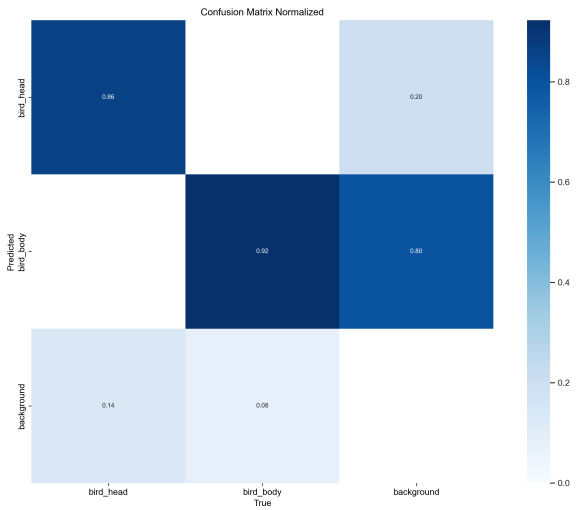

Fig. 6. Confusion matrix of bird parts classification

The F1-curve (fig. 7), illustrates how the model's F1-score changes as the confidence threshold increases. The curve shows that the model performs best within a specific threshold range, where it maintains a good balance between precision and recall.

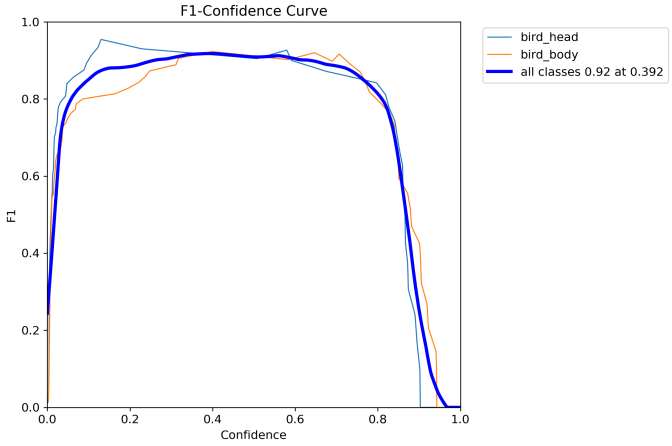

Fig. 7. F1-score by confidence curve

2) Detecting bird's head and body of the entire dataset: Once the model has been trained and validated, it can now detect the heads and bodies of birds in all the images in the dataset. A confidence threshold of 50% is used so as not to limit detection too much, especially given that the training was done with a small proportion of examples.

2 files were then saved, or 2 datasets, one with images of the bird's head, and the other with images of the bird's body, detected with the previously trained model. It should be noted that a bounding box is obtained, meaning that in some images, some of the background is still visible. Below are some examples of the images present in the two datasets:

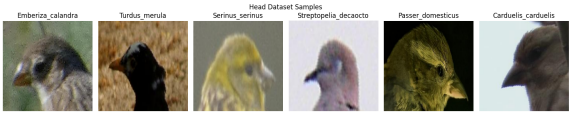

Fig. 8. Examples of samples detected by the YOLO model, head

Fig. 9. Examples of samples detected by the YOLO model, body

Although the examples above seem quite accurate, some detections are not so good, with the head sometimes appearing when detecting the bird's body, or sometimes even parts of the background being detected as if they were part of the bird's body. On the other hand, in some images, there was no detection at all, mainly due to the confidence threshold. Even so, two datasets with several samples were obtained. Below is the distribution of images according to their species and body part:

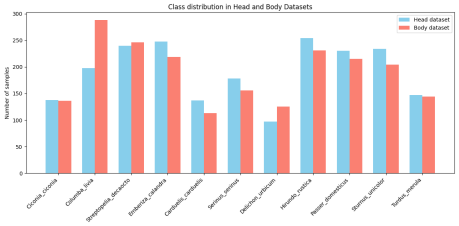

Fig. 10. Dataset Samples Distribution

Finally, the two datasets are put through the data augmentation process, making changes to the images such as rotation, or changes to the color itself, and also preparing the images for the pre-trained models.

3) Head Model Training: The head model was trained using cropped images of the bird's head.

The model achieved the following scores:

- Macro F1-score: ~ 0.86
- Accuracy: ~ 0.87
- Macro-AUPRC: ~ 0.93
- Top-3 Accuracy: ~ 0.96

The head-based classifier demonstrates strong and consistent performance across most species. The matrix (fig. 11) shows that predictions are more concentrated along the diagonal, indicating reliable class separation. While some confusion still exists — particularly between visually similar species — the head model maintains more precise decision boundaries.

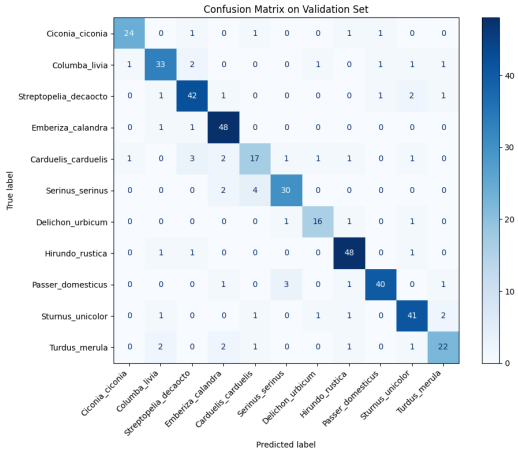

Fig. 11. Confusion matrix, head classification, 11 classes

It is also interesting to observe the Grad-Cam of this model (fig. 12), especially as it is a model that aims to better distinguish between similar species by analyzing the patterns and details of this part of the body.

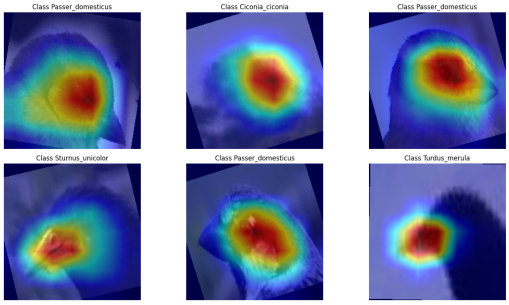

Fig. 12. Grad-CAM, head examples

As can be seen in these examples, the model focuses a lot on the eye and the beak, whereas it distinguishes only by the bird's head.

4) Body Model Training: The body model training follows the same methodology as the head model. Features such as body pattern, wing structure, and plumage contrast are expected to play a central role in this model's performance.

The model achieved the following scores:

- Macro F1-score: ~ 0.85
- Accuracy: ~ 0.86
- Macro-AUPRC: ~ 0.92
- Top-3 Accuracy: ~ 0.96

The body classifier performs adequately but displays more dispersion in its predictions. Although species like Emberiza calandra and Hirundo rustica still show strong true positive counts, other classes like Carduelis carduelis are more frequently confused with one another.

Misclassifications are more scattered across the matrix (fig. 13), implying that body-only features may not always provide enough discrimination, especially for species with similar body shapes or plumage.

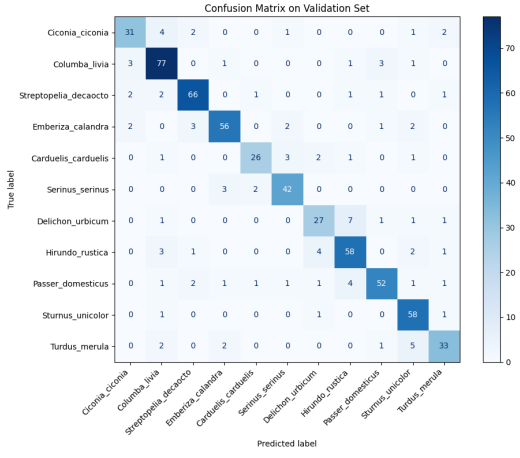

Fig. 13. Confusion matrix, body classification, 11 classes

It is also interesting to observe the Grad-Cam of this model (fig. 14), especially as it is a model that aims to better

distinguish between similar species by analyzing the patterns and details of this part of the body.

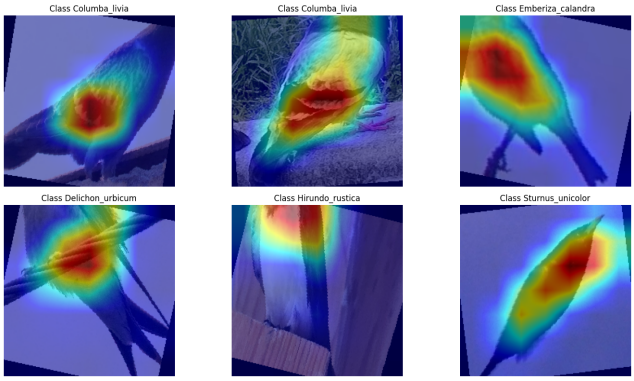

Fig. 14. Grad-CAM, body examples

In the case of the model trained to differentiate between birds' bodies, it is possible to see that the model focuses on certain body patterns, or even shape, as in the case of the last example.

D. Ensemble

For this last part of the project, it was decided to put together the models trained with the whole image, trained with the head and trained with the body. In principle, this will produce better results.

Firstly, a BirdClassifierEnsemble class (BirdClassifier.py file) was created, with functions to support image classification. This class implements a bird species classification system by combining the 3 models. It uses the YOLOv8 model, trained before, to detect and crop the head and body regions from the input image, which are then classified separately. In addition to these, the model trained on the full image is also used. The predictions from these models are aggregated using different strategies such as mean, max, or voting, and the final output returns the top-k most probable species along with their confidence scores.

Then, using all the images in the dataset, we classified them all by averaging the outputs obtained by the 3 models and obtained the results.

This classification achieved the following scores:

- Macro F1-score: ~ 0.92
- Accuracy: ~ 0.92
- Macro-AUPRC: ~ 0.96
- Top-3 Accuracy: ~ 0.98

As can be seen in the confusion matrix 15, most of the species can be well distinguished. Despite this, in some species, such as Delichon urbicum and Hirundo rustica, there is still some difficulty in distinguishing them, mainly because only the model trained with images of the head can make this distinction better.

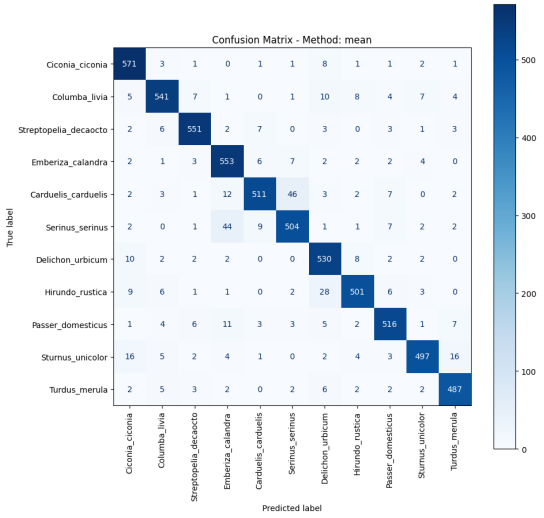

Fig. 15. Confusion matrix, 11 classes, ensemble classification

ID	Model	Input Type	Description	Macro-F1	Accuracy	Macro-AUPRC	Top-3 Acc	Notes
Base	EfficientNet-V2-s	Full Image	Multiclass baseline	~0.83	~0.83	~0.98	~0.95	Baseline
A	EfficientNet-B0	Head Crop	Multiclass trained on head crops	~0.86	~0.87	~0.93	~0.97	
B	EfficientNet-B0	Body Crop	Multiclass trained on body crops	~0.85	~0.86	~0.92	~0.96	
C	EfficientNet-B0 (Segmented)	3 equal vertical crops	Multiclass	~0.74	~0.75	~0.83	~0.91	Using YOLO + SAM crops
D	EfficientNet-B0 (Segmented)	3 equal vertical crops, One Vs All	Binary classifiers per species	~0.87	~0.88	~0.94	–	Using YOLO + SAM crops
E	Ensemble (full + head + body)	Combined logits	Fusion of A + B + Base	~0.92	~0.92	~0.96	~0.98	Late fusion

Fig. 16. Table Results

E. Table Results

IX. Contribution of each member of the group

- **Duarte** - researched and built the dataset, trained and improved the base model (full images)
- **João** - trained and improved the segmented model and part-based model, formed the respective datasets from the original dataset, and made the ensemble classification.

X. Conclusions

This project explored various deep learning strategies for classifying common bird species in Portugal, including full-image classification, segmentation-based modeling, and part-based models focused on the bird's head and body. Through rigorous experimentation with EfficientNet architectures and a custom-trained YOLOv8 detector, we demonstrated that combining multiple visual perspectives enhances classification performance. While each model showed unique strengths, the

ensemble approach achieved the most robust and accurate results across all metrics.

REFERENCES

[1] Merlin Bird ID, https://merlin.allaboutbirds.org/photo-id/
[2] Wilder, These 10 birds are among the most seen (and heard) in spring https://wilder.pt/primavera/estas-10-aves-estao-entre-as-mais-vistas-e-ouvidas-na-primavera
[3] EfficientNet, https://pytorch.org/vision/main/models/efficientnet.html
[4] YOLO, https://yolov8.com/
[5] INaturalist, https://www.inaturalist.org/
[6] GBIF, https://www.gbif.org/