JOURNAL OF
EMERGING INVESTIGATORS

Article

Identifying shark species using an AlexNet CNN model

Kuber Sarwal[1], Erick Ruiz[2], Amit Sarwal[3]
[1] Ridge High School, Basking Ridge, New Jersey
[2] MathWorks, Cambridge, Massachusetts
[3] IFF, Basking Ridge, New Jersey

SUMMARY

Identifying marine life is vital for maintaining biodiversity and environmental health, but current methods are hindered by the need for time-consuming, labor-intensive manual observations. This research introduces a convolutional neural network (CNN) model designed to accurately identify shark species. Our primary goal was to overcome challenges posed by limited datasets through transfer learning and pre-trained models. We hypothesized that an AlexNet CNN model would achieve superior accuracy in classifying shark species, compared to other CNN architectures, conventional algorithms, and custom neural networks, especially within the constraints of a limited dataset. AlexNet's deep convolutional layers and hierarchical learning capabilities were expected to enable effective feature extraction and learning from the limited data. Despite the challenge of working with a smaller dataset—only 100 images per species versus the recommended 5,000 samples per class—AlexNet's ability to capture spatial hierarchies and patterns led to enhanced performance. Our investigation involved comprehensive experimentation and comparative analysis to validate this hypothesis, offering insights into optimal shark species classification in resource-constrained scenarios. The model was trained on a Kaggle dataset containing 1,400 images across 14 shark species. We employed AlexNet as a feature extractor, with fine-tuning steps to adapt the network to this dataset. Experimental results showed that our model, termed "SharkNet," achieved a 93% accuracy on the test set, surpassing conventional methods. This promising performance in distinguishing shark species could significantly aid marine biologists and ocean conservationists in monitoring and protecting these species.

INTRODUCTION

To keep marine ecosystems in balance, sharks, as apex predators, are essential (1). Accurate identification and classification of shark species are essential for understanding their ecological roles, population dynamics, and the success of conservation efforts (2). However, manual identification of sharks based on visual characteristics can be time-consuming, subjective, and error prone. Due to their highly mobile nature and diverse habitats, sharks can be challenging to study. As such, tracking them requires sophisticated technology like satellite tags or acoustic telemetry, which can be expensive and logistically challenging to deploy and maintain (3). Sharks may reside in deep-sea regions which poses challenges for data collection, limiting opportunities for observation and research on their behavior, physiology, and ecology (3).

Computer vision algorithms like convolutional neural networks (CNNs) hold great potential for marine biologists and conservationists. Automating this process through computer vision algorithms reduces reliance on manual identification, saving valuable time and resources (2). Additionally, deep learning models can help uncover subtle and unique visual features distinguishing different shark species, contributing to a deeper understanding of their morphological variations and evolutionary adaptations (2).

Recent research demonstrated the efficacy of machine learning-driven object detection models in distinguishing between shark species, specifically the use of CNN-based models (4). Leveraging accurately labeled, diverse, and well-normalized datasets, these CNN models exhibit excellent performance in identifying sharks from aerial imagery captured by drones (4). Likewise, this same prior study highlights the importance of carefully curated training data and appropriate network architectures in achieving robust detection and classification results (4). For instance, not all shark species are equally distinguishable, mirroring challenges faced by human observers in discerning subtle differences between similar species. Factors such as image resolution, environmental conditions, and object size significantly influence the performance of CNN-based models (4).

Specifically, transfer learning algorithms have emerged as a promising solution (2). CNNs effectively minimize the number of learnable factors (such as weights and biases) by utilizing weight sharing, pooling layers, and local connection (2). This allows for the accurate processing of little data (2). AlexNet, a CNN introduced in 2012, advanced the field of deep learning and computer vision (4). One of the primary advantages of AlexNet over previous models is its depth and complexity: it can extract a variety of information from images thanks to its three 56 linked layers and five convolutional layers. In order to address the vanishing gradient problem, AlexNet uses the Rectified Linear Unit (ReLU) activation function, which enables faster training and improved performance even on smaller datasets (4). Additionally, its use of dropout layers helped mitigate overfitting, making it more robust for complex tasks (4). In the ImageNet Large Scale Visual Recognition Challenge (ILSVRC), AlexNet outperformed earlier state-of-the-art models, obtaining a top-5 error rate that was notably lower (4). The capacity of AlexNet to automatically

https://doi.org/10.59720/24-013

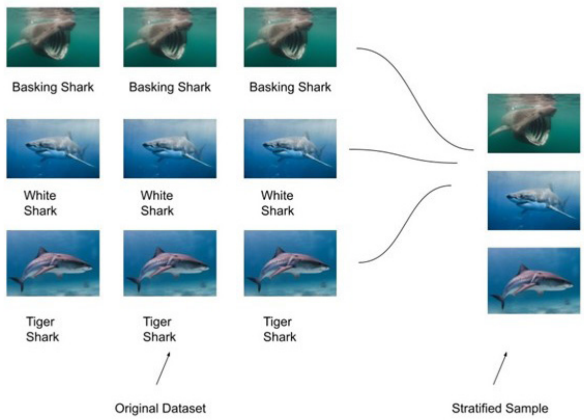

Figure 1: Documented model of a stratified shuffle split to diversify data. Stratified Shuffle Split is a cross-validation technique that ensures the distribution of class labels in the training and testing sets maintains the same proportion as the original dataset. It involves randomly shuffling and splitting the data while preserving the relative class frequencies.

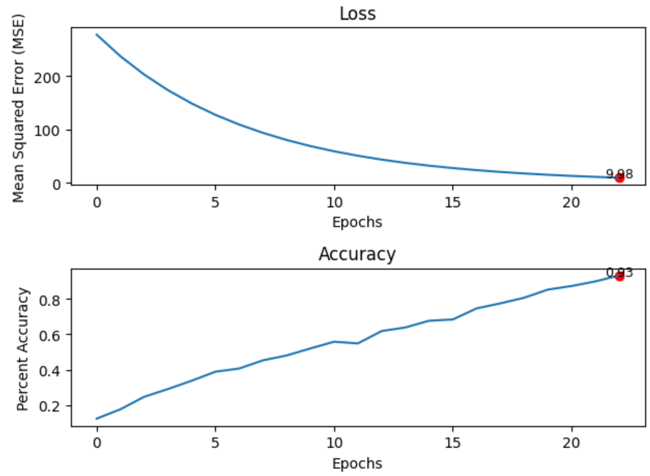

Figure 2: Plot of loss and accuracy graph after using AlexNet CNN model. The accuracy reached 93% after 23 epochs on the validation set. The loss reached 9.98 after 23 epochs on the validation set.

and adaptively learn spatial hierarchies of features makes it especially useful for jobs involving the interpretation of images and videos (4).

We aimed to address the challenge of identifying shark species using a limited dataset, a common issue in marine biology due to the difficulty in obtaining extensive data on sharks. We hypothesized that AlexNet's inherent ability to capture spatial hierarchies and patterns in image data would enhance its performance in shark species classification and help overcome the limitations posed by smaller datasets, unlike conventional CNNs. To test this hypothesis, we conducted a comparative analysis with other CNN models using a dataset obtained from Kaggle.com, consisting of 1400 images representing 14 different shark species (5). Kaggle is a website with many open-source datasets on various topics and is paired with downloadable csv files for the datasets (5). Leveraging the pre-trained weights of the AlexNet model, extensively trained on large-scale image datasets, we aimed to determine its efficacy in this context. Our results showed that AlexNet achieved a 93% accuracy on the test set for shark identification, demonstrating its potential for automated species identification in resource-constrained scenarios. These findings contribute to the growing field of computer vision applications in marine biology and suggest that deep learning models like AlexNet can significantly aid marine biologists in accurately identifying shark species with limited data availability.

RESULTS

Dataset Preparation and Initial Attempts

The dataset from Kaggle.com, consisting of 1,400 images of 14 shark species, was standardized to 150 x 150 pixels (5). Next, the dataset was split into training and testing sets with a split ratio of 66% for training and 33% for testing. We used a stratified shuffle split on the dataset to ensure an even distribution of images from each category (6). A stratified shuffle split is a data splitting technique that ensures each subset of data maintains the same proportion of classes as the original dataset, while randomly shuffling the data to create training and testing sets (**Figure 1**). This method increased the diversity of the test and training sets, which is especially crucial for small datasets where data may be easily skewed. Initially, we employed logistic regression but achieved only 23% accuracy. Subsequently, a custom neural network was developed, achieving a 9% accuracy.

Given the challenges with logistic regression and custom neural networks, we implemented the AlexNet CNN model. Several convolutional layers for feature extraction and fully linked layers for classification are included in AlexNet's design. During initial training stages, a notable decrease in accuracy was observed due to the difference in image sizes between the original ImageNet training (230x230 pixels) and our standardized size (150x150 pixels). To address this, we made several adjustments: changing the filter size in the first convolutional layer, modifying the kernel size, adjusting the MaxPooling pool size, removing the last convolutional layer, and adding l2 regularizers in the fully connected layers, which improved accuracy. After training for 23 epochs, the model achieved an accuracy of 93% on the test set and a mean squared error loss of around 11.61 (**Figure 2**). The AlexNet model achieved a consistent accuracy around 90% for each species. The highest accuracy was 97.8% for the basking shark, likely due to its high contrast in colors and abstract edges, while the lowest was 81.5% for the white shark. We generated a confusion matrix to better illustrate the AlexNet models true accuracy when identifying a number of different species (**Figure 3**). Further, a pie chart of prediction confidence provided deeper insights into model performance and misclassification patterns by showing the top 3 predictions for a random test case. This displayed high confidence in the models predictions as it shows as large difference in the most likely prediction probability and the others (tiger shark and whale shark probabilities) (**Figure 4**). Finally, a matrix was made where each cell displays the average confidence percentage, indicating how often the model assigns a particular confidence to each shark type relative to others. The most confident prediction was the whitetip shark and the least confident prediction was the bull shark (**Figure 5**). Most of the time the model defaulted to guessing the white shark. This could be because the model overfits to the noise and

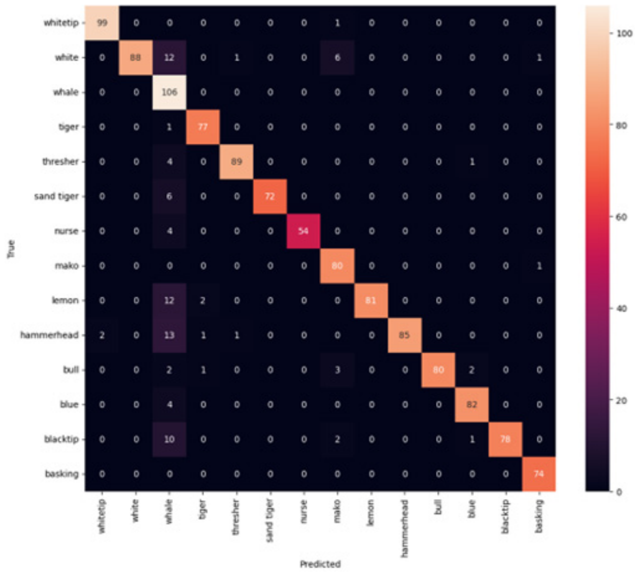

Figure 3: Confusion matrix of AlexNet CNN model. The x-axis represents the predicted species, while the y-axis shows the actual species. The heatmap displa`ys the count of predictions, with lighter colors indicating higher numbers. Diagonal cells represent true positives (TP), where predictions match the actual species, while off-diagonal cells show misclassifications, including false positives (FP) and false negatives (FN). The matrix reveals strong model accuracy, particularly along the diagonal, with some misclassifications such as "whitetip" being confused with "hammerhead" and "lemon" with "blacktip”.

color patterns of the white shark so it may default to predicting a white shark.

Generalization to Unseen Data

To evaluate the AlexNet model's robustness, we tested its performance on images sourced from the internet, achieving a 90.6% accuracy. These images were retrieved from google and adobe stock. These images displayed a variety of different species such as a white shark, nurse shark, and tiger shark. The same preprocessing was done as with the Kaggle images (resizing them to 150x150 pixels).

DISCUSSION

Using a dataset from Kaggle.com consisting of 1,400 images across 14 shark species, our AlexNet CNN model achieved a high accuracy of 93% after 23 epochs of training (5). Further, when tested on new, unseen images sourced from Adobe Stock, the model maintained a similar accuracy of 90.6%, confirming its ability to generalize well to new data and its practical applicability in marine biology. The success of our model in identifying shark species marks a step forward in the application of deep learning to this field.

Other studies have also explored the use of deep learning for species identification. For example, studies like Mohanty et al. applied conventional CNNs for identifying various marine species, achieving comparable accuracy rates (8). This study reached an accuracy of 83% on the validation set after training for 12 epochs with the Adam optimizer and categorical cross entropy loss function (7). However, the Mohanty study had access to larger and more varied datasets (14,346 training images compared to our 1400 images). Our study aligned with

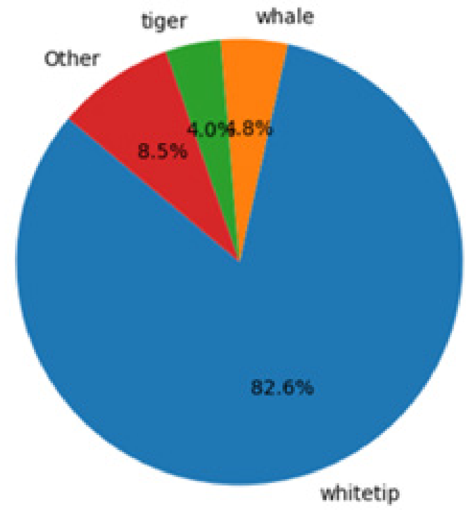

Figure 4. Confidence distribution of top three predictions for each shark species. Each pie chart represents a test image, showing the proportion of confidence attributed to the top three predicted species, with the remaining confidence aggregated into an "Other" category. The charts highlight the model's certainty in its classifications and indicate which species are often confused with others, providing insights into the model's performance and areas for improvement.

their findings in demonstrating the effectiveness of CNNs, but specifically the scope of the AlexNet CNN for shark species classification.

The implications of our findings extend beyond shark species identification. The demonstrated effectiveness of the AlexNet model in this domain highlights the potential for similar deep learning approaches to be applied in other areas of marine biology, such as identifying different marine organisms like dolphins or other fish species. In conclusion, this study highlights the value of deep learning approaches in marine biology research and conservation. The AlexNet model’s high accuracy in identifying shark species from images not only aids in ecological monitoring but also exemplifies the broader applicability of CNNs in image classification problems (8). As automated species identification methods continue to develop, they will significantly enhance our ability to preserve and understand marine ecosystems. This emerging methodology of utilizing transfer learning techniques for animal classification holds great promise for improving marine conservation efforts and advancing our knowledge in the field.

Despite these promising results, our study had several limitations. One major constraint was the relatively small size of the dataset, which includes only 100 images per shark species. The model's capacity to generalize to novel, unseen images may be hampered by this restriction. However, when the model was tested on new images from the internet, the accuracy was maintained which demonstrates that the limited dataset was not a crucial limitation. Additionally, the diversity of the dataset in terms of environmental conditions and shark poses is limited and may not have fully capture the variability encountered in real-world scenarios. Improving the model may involve fine-tuning its architecture and parameters. For instance, modifications such as changing the kernel sizes and

https://doi.org/10.59720/24-013

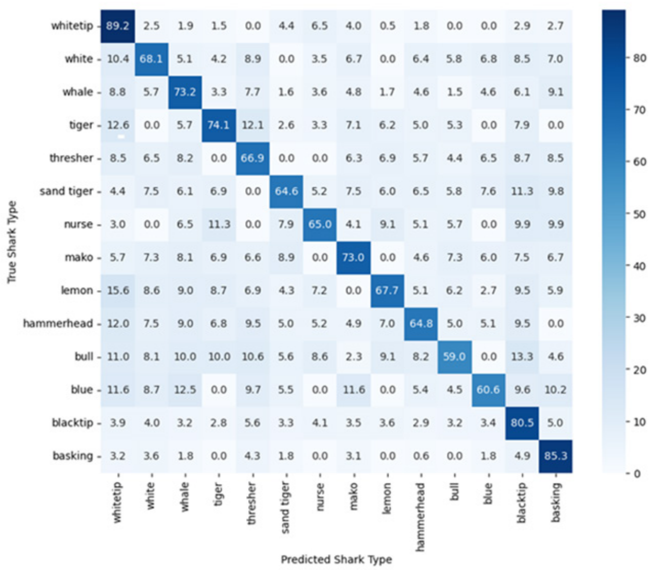

Figure 5. Average confidence of predictions by shark type. The x-axis represents the predicted species, while the y-axis shows the actual species. Each box represents the average % confidence that the algorithm had in its prediction. A darker color signifies a more confident predictions whilst a lighter color means a less confident one. Lemon sharks are the most confused to be a white tip shark, and the mako shark is most confused to be a sand tiger shark. The most confident predictions come from the whitetip shark.

pooling layers were implemented (e.g., adjusting the input shape to 150x150 pixels and the MaxPooling pool size to (2,2)). However, these optimizations were constrained by the scope of this study and the computational resources available (9). Exploring ensemble methods or combining multiple pre-trained models could also potentially improve classification performance. Implementing a validation set in the training-test split was one step taken to prevent overfitting and aid in hyperparameter tuning, but more sophisticated approaches like cross-validation could be employed for better model validation (10).

Future work could expand the scope of this study by incorporating larger and more diverse datasets. Collaborating with marine biologists to gather more comprehensive shark image data, including rare species and various environmental conditions, would be beneficial. Additionally, applying advanced augmentation techniques to artificially increase the diversity of the training data could help in making the model more robust. Further research could also explore the integration of multi-modal data, such as combining visual data with acoustic or environmental data, to enhance species identification. Incorporating temporal data to track shark movements and behaviors over time might also provide deeper insights and improve model accuracy.

MATERIALS AND METHODS

Dataset

The image dataset used in this research consisted of shark images obtained from Kaggle.com (5). The dataset contains a total of 1,400 images, with 100 images for each of 14 different shark species. The shark species are: basking, blacktip, blue, bull, hammerhead, lemon, mako, nurse, sand tiger, thresher, tiger, whale, white, and whitetip. The photos

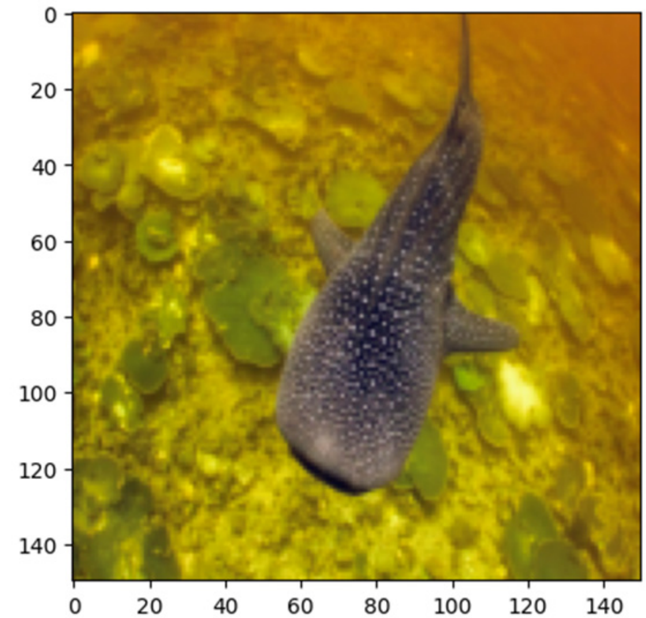

Figure 6: Sample from the nurse shark image dataset. Using matplotlib, we resized each image to a uniform 150x150 pixel size so the inputs of the neural network would be consistently sized for each image and faster to process (7).

were taken in a variety of environmental settings, including different lighting, water clarity, and viewing angles, aiming to capture the natural variations in shark appearances (**Figure 6**).

To prepare the dataset for training the AlexNet CNN model, several preprocessing steps were applied. The dataset from Kaggle.com, consisting of 1,400 images of 14 shark species, was standardized to 150 x 150 pixels. Originally, the images ranged from 500 x 347 pixels to 1200x758 pixels large. Next, the dataset was split into training and testing sets with a split ratio of 66% for training and 33% for testing. We used the NumPy random.permutation function to create new randomly generated lists of the dataset and sklearn's StratifiedShuffleSplit function to evenly distribute the data for the training, validation, and test sets.

The training set, consisting of 923 images (66%), was used to optimize the model's parameters and learn the underlying representations. The testing set, comprising of 476 images (33%), served as an independent dataset for evaluating the model's performance in generalizing to unseen shark images. To visually represent the performance of the AlexNet CNN model, I utilized Python libraries Matplotlib and Seaborn to generate the necessary figures. After training and saving the compiled model, I used these libraries to create detailed graphs, including the confusion matrix and loss/accuracy curves. The confusion matrix was plotted to show the model's classification accuracy across different shark species, while the loss and accuracy graphs were generated to illustrate the model's performance over the training epochs. These visualizations were crucial in understanding and interpreting the model's effectiveness and areas for improvement.

Classification Models

We first implemented a logistic regression model using the LogisticRegression class from the sklearn.linear_model

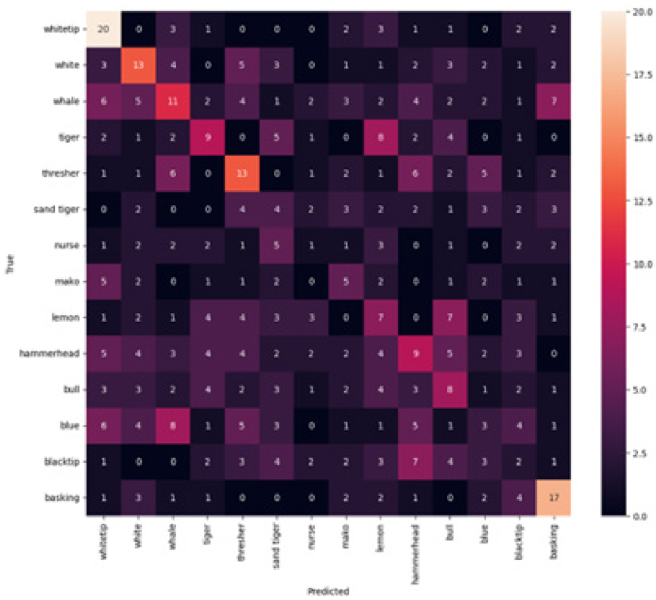

Figure 7: Confusion matrix of logistic regression model. The x-axis represents the predicted species, while the y-axis shows the actual species. The heatmap displays the count of predictions, with lighter colors indicating higher numbers. Diagonal cells represent true positives (TP), where predictions match the actual species, while off-diagonal cells show misclassifications, including false positives (FP) and false negatives (FN). The matrix reveals weak model accuracy with some misclassifications such as "nurse" being confused for "mako".

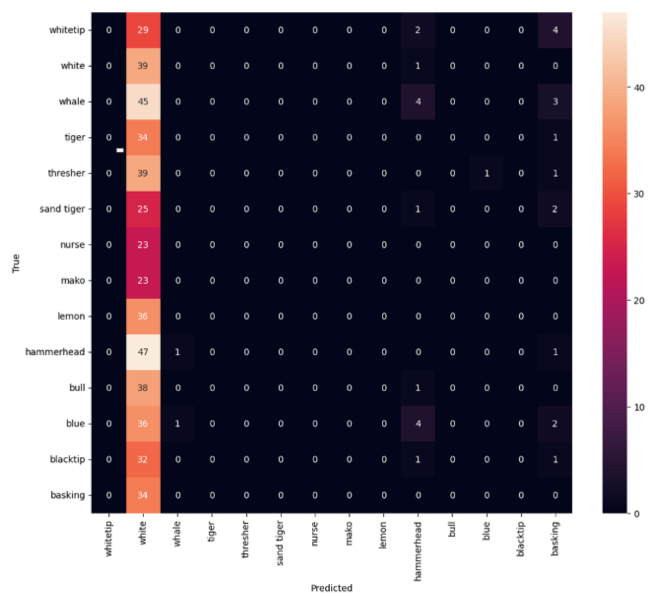

Figure 8: Confusion matrix of custom neural network model. The x-axis represents the predicted species, while the y-axis shows the actual species. The heatmap displays the count of predictions, with lighter colors indicating higher numbers. Diagonal cells represent true positives (TP), where predictions match the actual species, while off-diagonal cells show misclassifications, including false positives (FP) and false negatives (FN). The matrix reveals weak model accuracy with some misclassifications such as "hammerhead" being confused for "whale".

module. The model was trained on the training dataset (X_train and y_train) with random_state set to 0 for reproducibility. After fitting the model, we predicted the shark species on the test dataset (X_test). To evaluate the model's performance, we calculated the accuracy using sklearn.metrics and generated a confusion matrix using a seaborn heatmap with matplotlib to visualize the distribution of predictions (**Figure 7**).

After testing a logistic regression, a CNN was implemented using the sequential model from the Keras library (11). The model architecture consisted of several layers designed to utilize filters to extract distinctive features from the image. It began with three convolutional layers that progressively extract features. The first convolutional layer used 150 filters with a kernel size of 3x3 and applies 'same' padding. The second convolutional layer had 100 filters, also with a 3x3 kernel size and 'same' padding. The third convolutional layer included 50 filters with the same kernel size and padding configuration.

After feature extraction, the output from the convolutional layers was flattened into a single-dimensional array to prepare it for the fully connected layers. The input layer was defined with an input shape of (150, 150, 3) to match the dimensions of the input images. A dense layer with 5 neurons and a ReLU activation function was added to the model, enabling it to learn complex patterns from the flattened feature map. The final output layer contains 14 neurons and uses a softmax activation function to classify the input images into one of the 14 shark species.

The model was constructed with the Adam optimizer for training, categorical cross-entropy as the loss function, and accuracy as the evaluation metric. The model architecture and hyperparameters were chosen based on initial experimentation and fine-tuning to achieve the best performance for the shark species classification task. Similar to the logistic regression model, to evaluate the model's performance, we calculated the accuracy using sklearn.metrics, and generated a confusion matrix using a seaborn heatmap with matplotlib to visualize the distribution of predictions (**Figure 8**).

Finally, the AlexNet CNN model was implemented using the Keras library (11). The model architecture started with a series of convolutional layers designed for feature extraction. The first convolutional layer employed 96 filters of size 5x5, with a stride of 2 and ReLU activation. This layer was followed by Batch Normalization and MaxPooling to reduce dimensionality and improve stability. The second and third convolutional layers each utilized 236 filters of size 5x5 with 'same' padding, also followed by Batch Normalization and MaxPooling.

After the convolutional layers, the output was flattened to serve as input for the fully connected layers. The model includes two dense layers, each with 1023 units and ReLU activation. These layers were regularized using L2 regularization and incorporate Dropout layers with a rate of 0.5 to prevent overfitting. The final output layer was a dense layer with 14 units and softmax activation, providing a probability distribution across the 14 shark species classes.

The Adam optimizer and the categorical cross-entropy loss function were used to compile the model, ensuring efficient training and accurate classification. For more details on the code behind the models, see the git repository made for this project (12).

https://doi.org/10.59720/24-013

ACKNOWLEDGMENTS

We would like to acknowledge and give our warmest thanks to the Inspirit AI program for helping to instruct the entire research process. Without his help, none of this would be possible.

Received: December 10, 2023
Accepted: June 5, 2024
Published: September 23, 2024

REFERENCES

1. Kelli. "Why Are Sharks Important?" Save Our Seas Foundation, 11 Nov. 2020, www.saveourseas.com/why-are-sharks-important/#:~:text=Alongside%20other%20predators%2C%20sharks%20play. Accessed 21 May 2023.
2. Shark Trust. "Shark Taxonomy." The Shark Trust, 6 Nov. 2018, www.sharktrust.org/shark-taxonomy#:~:text=It. Accessed 21 May 2023.
3. "Shark Finning and Shark Fin Facts." Shark Stewards, 18 Oct. 2018, sharkstewards.org/shark-finning/shark-finning-fin-facts/#:~=By%20nature%2C%20sharks%20are%20difficult. Accessed 21 May 2023.
4. Purcell C, Williamson JE, Le Port A, Wilson T, Humphries N, Smale M. "Assessing the Ability of Deep Learning Techniques to Perform Real-Time Identification of Shark Species in Live Streaming Video from Drones." *Frontiers in Marine Science*, 20 Oct. 2022, https://doi.org/10.3389/fmars.2022.981897.
5. Lautaro. "Shark Species." Kaggle, www.kaggle.com/datasets/larusso94/shark-species. Accessed 21 May 2023.
6. Oliphant T. "NumPy User Guide — NumPy v2.0.dev0 Manual." Numpy.org, numpy.org/devdocs/user/index.html#user. Accessed 21 May 2023.
7. Burkhardt D. "The Impact of Transfer Learning and Fine-Tuning on CHATGPT's Conversational Performance." Saturn Cloud Blog, 4 Oct. 2023, saturncloud.io/blog/the-impact-of-transfer-learning-and-fine-tuning-on-chatgpts-conversational-performance/. Accessed 21 May 2023.
8. Krizhevsky A, Sutskever I, Hinton GE. NeurIPS, proceedings.neurips.cc/paper_files/paper/2012/file/c399862d3b9d6b76c8436e923a68c45b-Paper.pdf. Accessed 21 May 2023.
9. Mohanty, Anishka, Guillermo Goldsztein, and Raphaël Pellegrin. "Fish Species Image Classification Using Convolutional Neural Networks." *Journal of Student Research*, vol. 11, no. 3, 31 August 2022, pp. 1–12, https://doi.org/10.47611/jsrhs.v11i3.3058.
10. Soper, Daniel S. "Greed Is Good: Rapid Hyperparameter Optimization and Model Selection Using Greedy k-Fold Cross Validation." *Electronics*, vol. 10, no. 16 August 2021, pp. 1-30, https://doi.org/10.3390/electronics10161973.
11. Team Keras. "Keras Documentation: The Sequential Class." Keras.io, keras.io/api/models/sequential/. Accessed 21 May 2023.
12. Sharko123. "Shark Data Exploration." GitHub, 19 December 2023, github.com/Sharko123/SharkDataExploration.git.