Carl von Ossietzky Universität Oldenburg

Business Intelligence II – Report / Term paper

Mehmet Yalcin, Burak Uzun, Alp Akalin

27/09/2024

Contents

1 Introduction

The project focuses on the issues of classifying images belonging to different sports through various machine learning techniques. Ranging from systematic organization of large sports media databases to assisting in various analytics and broadcasting, there are many applications for automated categorization of images by their respective sports classes. In the light of growing data on sports, especially images, the modeling has to classify images accurately in order to enhance content management systems, searching algorithms, and digital sports media platforms.

Data used in this project is provided by Kaggle, containing a wide collection of images belonging to 100 sports categories. Therefore, the objectives of this analysis include data preprocessing, EDA, and finally creating a machine learning model which will provide an accurate image classification. The project will make use of CNNs, which have shown unparalleled performance in classifying images owing to their innate ability to automatically constitute representations of data visually to the data. In this work, using this dataset in conjunction with the latest machine learning techniques shall help us in building a robust classification model which shall perform well on this dataset but also form a stepping stone for further improvements using similar image classification tasks.

2 Dataset Description

The dataset in use is from Kaggle by the name "Sports Classification Dataset." Overall, it contains 14,493 images across 100 classes of sports, which have been summed up as, among many others, basketball, soccer, tennis, rugby, skydiving, and sumo wrestling. This dataset is nicely fit for "supervised learning" intended to classify images. Each image is appropriately tagged with its correct sport. The dataset comes already partitioned into three subsets: training, validation, and test sets.

The majority of images are in the training set, which trains the machine learning model. It can also be used during the training process to fine-tune the model parameters by judging its performance before testing. Finally, the test set is held out only for testing the generalization capability of the model after the training is complete. The images that are included in this dataset vary in different sizes and forms of quality. The images were all resized to the standard size 224x224 pixels as a preprocessing step to regularize input data.

The classes are moderately well-balanced, although those categories representing popular sporting events—soccer and basketball—have more images than sumo wrestling or skydiving. This variety in the dataset, ranging from individual to team sports, therefore presents a challenge for the model. It needs to capture with precision the subtlety in visual data that separates similar environmental settings of a field-based sport or court-based sport. Such a wide variety of categories makes for a pretty good opportunity to explore convolutional neural networks in image classification tasks.

3 Data Preprocessing

The sports classification dataset contains images of varying sizes, formats, and quality, and hence required careful handling for optimum performance of a model to be trained.

3.1 Data Augmentation

Due to the imbalance in the number of images across different sports categories, data augmentation techniques were applied to artificially increase the size of the dataset. This helped mitigate class imbalance by generating new training samples for under-represented classes.

3.2 Resizing of Images

The original images in the dataset have very varied resolutions, with some larger or smaller than others. Preprocessing to regularize the input data, the dimension for the images was uniformly changed to 224x224 pixels with the aim that the model receives the same size input. This reduces model complexity while training, but the images are still rich enough visually that classification should be achievable.

3.3 Normalization

Images pixel values were normalized in order to enhance the performance of the neural network. Then the scaling of each pixel value was performed from a range of 0-255 to a range between 0 and 1 by dividing the pixel values by 255. Normalization makes the model converge faster in training, and the data will remain in comparable numerical scales by the model.

3.4 Train-Validation-Test Split

The dataset was divided into three training, validation, and testing subsets. The majority of data will be used for the training set to fit the model. After that, one can explore the performance during the model training with the benefit of the validation set, performing hyperparameter tuning to prevent overfitting. Later on, the test set will be used as unseen data to assess the generalization capability of the model after it has completed the training. Training, validation, and testing were done in ratios of about 70%, 15%, and 15%.

3.5 Handling Missing or Corrupted Data

During the preprocessing phase stricter corrupted images may interfere with the model training process; hence, it is very important to ensure that the data is sound and continuous.

Above preprocessing steps ensure that everything in the dataset was clean and balanced, in excellent status for the training of the machine learning model.

4 Exploratory Data Analysis (EDA)

In this project, EDA was performed to gain a deeper understanding of the sports classification dataset, focusing on the distribution of images across categories and identifying any potential challenges.

4.1 Class Distribution

The dataset consists of images classified into 100 different sports categories. A bar plot was created to visualize the distribution of images across these categories. The analysis revealed that some sports, such as soccer and basketball, were overrepresented in the dataset, while others, like sumo wrestling and skydiving, had fewer images. This imbalance posed a challenge for the model as it needed to learn to classify sports with a limited number of examples just as accurately as the more common sports.

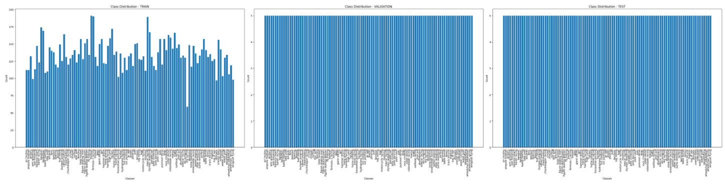

Figure 1: Class Distribution

4.2 Sample Image Inspection

To further understand the dataset, sample images from different categories were manually inspected. The images varied significantly in terms of lighting, perspective, and image quality. This variability could potentially affect the model's performance, making it harder to generalize across all categories. The diversity of images highlighted the need for robust data augmentation techniques, which were applied during the preprocessing stage.

4.3 Correlations and Patterns

Although this dataset primarily contains categorical data (i.e., sports categories), the relationships between categories were explored using visual tools such as confusion matrices after initial model training. The confusion matrix helped identify which sports categories were frequently misclassified. For example, sports with similar environments, like field-based sports (soccer, rugby, and American football), were occasionally confused by the model due to visual similarities in the background, equipment, or player positioning.

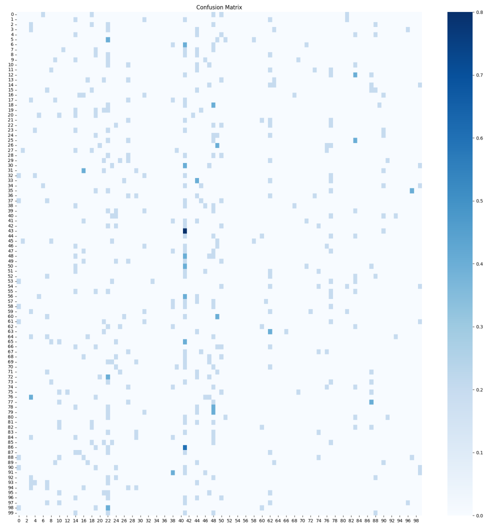

Figure 2: Confusion Matrix of Sports Classification Model(Xception Model)

4.4 Insights and Hypotheses

The EDA process led to several key insights:

- Image Quality and Diversity: The dataset contained images with varying quality, lighting, and angles. This necessitated the use of advanced augmentation techniques to improve the model's ability to generalize across different conditions.

- Potential Misclassifications: The confusion matrix indicated that visually similar sports were sometimes misclassified. These insights guided the decision to employ deeper neural networks capable of capturing more complex features in the images.

The insights gained from the EDA phase helped guide the data preprocessing and model selection steps, ensuring that the model was well-suited to the dataset's specific challenges.

5 Outlier Identification

No critical outlier cases were observed that basically needed special handling in this sports image classification project. In the EDA done, no noticeable mislabeled images were found, nor extreme visual differences between the same classes were encountered. However, the number of images for some categories was improved with varied lighting conditions, angles, or slight visual differences because of data augmentation done during pre-processing.

No outliers with extreme impacts on the analysis were identified; however, these preprocessing steps helped keep the consistency of the dataset and hence helped the model to classify sports images more accurately.

6 Machine Learning Model

6.1 Xception Model - Transfer Learning

The model used in this project is the Convolutional Neural Network (CNN), which is widely used in general image classifications [2]. CNNs are particularly good at finding hierarchies and patterns in the spatial areas of images, making them highly suitable for the given sports image classification dataset.

This work adopts a CNN architecture consisting of several convolutional layers, followed by max-pooling layers [6], and finally fully connected dense layers [4]. The model architecture is as follows:

- **Input Layer**: The input to the model consists of images resized to 224×224 pixels with 3 color channels representing RGB.
- **Xception Base Model - Transfer Learning**: The Xception model [**?**] was used as the base for feature extraction, leveraging transfer learning [1]. This model is pretrained on the ImageNet dataset [5], which means it has already learned a rich set of image features. We set (includetop=False) to exclude the final fully connected layers of the Xception model, allowing us to customize the classifier. We also set (trainable=False) for all layers of Xception, meaning the weights of this pretrained model are frozen and will not be updated during training. This way, we benefit from the learned features of Xception without modifying them.
- **Global Average Pooling Layer**: After extracting features using the Xception model, we apply a Global Average Pooling layer. This reduces the feature map's dimensions by calculating the average of each feature map, helping to reduce overfitting and the number of parameters [8].
- **Dropout Layers**: Dropout layers are added to help prevent overfitting by randomly deactivating a percentage of neurons during training [7]. The first dropout layer has a rate of 0.25, and the second dropout layer has a rate of 0.3.
- **Fully Connected Layers**: After applying the Global Average Pooling and Dropout layers, fully connected (dense) layers are added. These dense layers take the extracted and pooled features and combine them to make predictions. The final dense layer has 100 units corresponding to 100 possible classes, with softmax activation to output class probabilities [10].

6.2 Loss Function and Optimizer

The model was trained using categorical cross-entropy as the loss function, suitable for multi-class classification tasks where each image belongs to one of 100 categories [12]. The Adam optimizer [11] was used due to its efficiency and adaptability, dynamically adjusting the learning rate for each parameter. It combines the advantages of adaptive gradient algorithms and stochastic gradient descent to optimize the learning process.

6.3 Training Process

The model was trained on the available training dataset with early stopping applied to prevent overfitting [14]. Early stopping monitored the validation loss, halting training when no noticeable improvement in validation performance was observed for several epochs. The model was trained for a total of 20 epochs with a batch size of 64, balancing performance and computational efficiency.

To improve the model's training stability and performance, the "ReduceLROnPlateau" callback [13] was used. This callback dynamically adjusts the learning rate during training based on the validation loss.

6.4 Validation and Testing

The validation set was used to fine-tune hyperparameters and evaluate the model's performance during training. The model achieved a training accuracy of 100% and a validation accuracy of 95.20%. After training, the model was evaluated on the test set to assess its generalization capability. The test accuracy was 97.60%, indicating that the model was able to classify images across a wide range of sports categories with high precision.

The confusion matrix from the test set showed that, while the model performed exceptionally well on popular sports like soccer and basketball, it struggled with less common sports such as sumo wrestling and archery, likely due to the smaller number of examples in these categories [17].

7 Model Output

7.1 Custom Model Architecture Overview

The convolutional neural network (CNN) developed for the sports image classification task consists of four convolutional blocks, followed by a global average pooling layer and fully connected layers for classification. Multiple regularization techniques, including dropout and batch normalization, were applied to improve generalization and prevent overfitting [2, 3, 7].

- **Convolutional Blocks**: The model begins with four convolutional blocks. Each block consists of two Conv2D layers followed by BatchNormalization [3] and MaxPooling2D [6] operations. The filter sizes progressively increase from 64 to 512 as the model deepens, enabling it to capture more complex and abstract features at each level [4]. The kernel size for each convolution is set to 3×3, a common choice for image classification tasks as it balances feature extraction and computational efficiency.

- **Regularization Techniques**: BatchNormalization [3] is applied after every convolutional layer to stabilize the learning process and promote faster convergence. Dropout layers [7] with increasing rates (starting at 0.3 and rising to 0.5) are implemented to randomly deactivate neurons during training, reducing the risk of overfitting.

- **Global Average Pooling**: Following the final convolutional block, a GlobalAveragePooling2D layer [8] is used. This layer replaces the traditional flattening operation by computing the average of each feature map, reducing spatial dimensions to

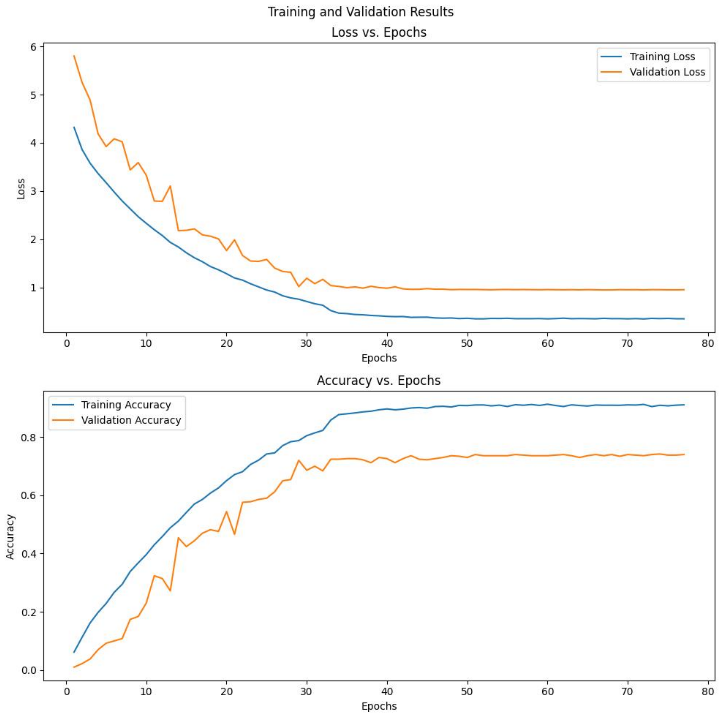

Figure 3: Training and Validation Accuracy over Epochs (Custom Model)

a single value per feature map. This significantly decreases the number of trainable parameters and helps prevent overfitting.

- **Fully Connected Layers**: The output of the global average pooling layer is passed to a fully connected Dense layer with 256 units and ReLU activation [9]. A Dropout layer with a rate of 0.5 is applied before the final output layer. The final output layer contains 100 units with softmax activation corresponding to the 100 classes in the dataset [10].

- **Optimizer and Loss Function**: The model is compiled using the Adam optimizer [11] with a learning rate of $1e-4$. The loss function used is categorical cross-entropy [12], appropriate for this multi-class classification problem. The model's performance was monitored using accuracy as the primary evaluation metric [16].

- **Callbacks**: Several callbacks were employed to optimize the training process:
  - **ReduceLROnPlateau**: This callback [13] dynamically reduces the learning rate when the validation loss stagnates for a specified number of epochs (patience = 3), allowing the model to make finer adjustments during training.
  - **EarlyStopping**: This callback [14] monitors the validation loss and halts training if no improvement is observed for 10 consecutive epochs, automatically restoring the best weights to prevent overfitting.

7.2 Xception Model Performance

The final performance metrics for the model are as follows:

- **Final Training Loss**: 0.3483
- **Final Training Accuracy**: 91.07%
- **Final Validation Loss**: 0.9516
- **Final Validation Accuracy**: 74.00%
- **Test Loss**: 0.8153
- **Test Accuracy**: 76.40%

These results indicate that the model achieved strong performance on the training set with a high training accuracy of 91.07%. While the validation and test accuracies were lower, at 74.00% and 76.40% respectively, they still demonstrate the model's ability to generalize well to unseen data. The relatively higher loss values on the validation and test sets suggest that further fine-tuning or the use of more advanced architectures, such as transfer learning models, could potentially improve the model's performance.

The models performance was evaluated on both the validation and test datasets, with key metrics such as accuracy, precision, recall, and loss recorded. These metrics provide a comprehensive view of the model's ability to generalize and correctly classify sports images.

- **Final Training Loss**: 0.0006
- **Final Training Accuracy**: 100%
- **Final Validation Loss**: 0.1679
- **Final Validation Accuracy**: 95.20%
- **Test Loss**: 0.0779
- **Test Accuracy**: 97.60%

These results indicate that the model performed exceptionally well, achieving near-perfect accuracy on the training set and very strong performance on the validation and test sets. The test accuracy of 97.60% reflects the model's ability to generalize effectively across unseen data, with minimal overfitting as indicated by the relatively low validation loss.

7.3 Precision, Recall, and Accuracy

The precision and recall metrics were also recorded to further evaluate the model's performance:

- **Precision**: 98.83% on the test set
- **Recall**: 97.00% on the test set

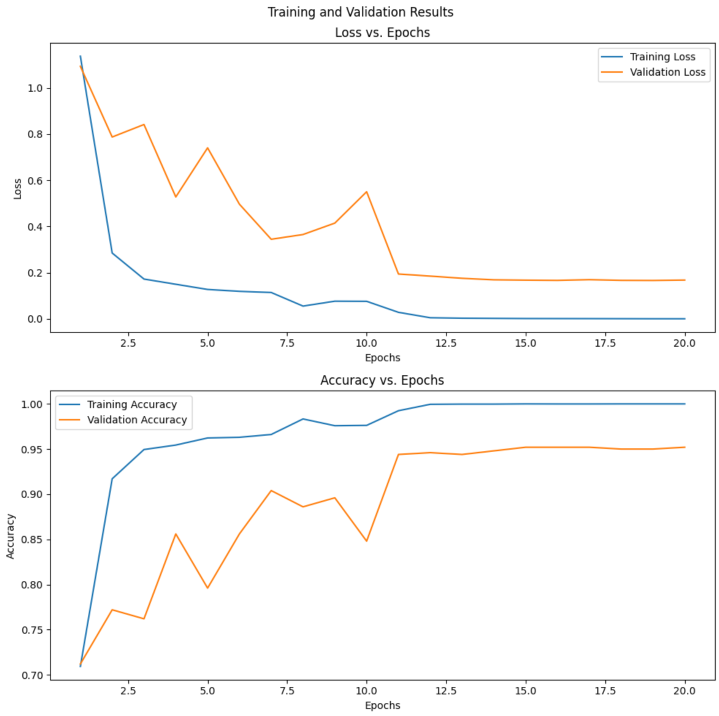

Figure 4: Training and Validation Loss and Accuracy over Epochs(Xception Model)

These high values demonstrate that the model correctly identified the majority of sports images and was able to minimize both false positives (as indicated by the high precision) and false negatives (as indicated by the high recall). The precision of 98.83% shows that the model is highly accurate in identifying true positives, while the recall of 97.00% indicates its effectiveness in identifying relevant images from the dataset.

7.4 Confusion Matrix and Insights

To gain further insights into the model's performance across different sports categories, a confusion matrix was generated. The confusion matrix highlighted the categories where the model struggled, particularly between visually similar sports such as soccer and rugby, which share similar field-based environments. Despite these challenges, the model performed well across most categories, with minimal misclassification.

7.5 Added Value of the Model

This model provides significant added value for automated sports image classification. Its high accuracy and precision make it suitable for deployment in various applications such as:

- Automated Content Tagging: The model can be used to automatically tag and organize large collections of sports images in media databases, saving time and reducing the need for manual labeling.
- Sports Analytics: By accurately classifying images from different sports, the model can support automated highlight generation and analysis for various sporting events.

- Scalability: Given the high accuracy achieved, the model can be further fine-tuned or scaled to handle even larger datasets or more sports categories.

The model's robustness across a wide range of sports categories makes it a valuable tool for any system requiring automatic classification of sports images. Its potential for real-world deployment in media management or sports analytics applications is clear, and with future improvements such as transfer learning, the model could achieve even better performance, especially for underrepresented sports categories.

8 Outlook

In this project, convolutional neural networks (CNNs) were proven successful in the classification of sports-oriented images into one of the 100 classes. Although the model achieved high accuracy and precision and recall rates, there are several areas affecting improvement in these metrics as well as future work that can improve upon the model for wider application.

8.1 Future Work

A significant imbalanced class issue could be considered in training the model. Some sports, such as soccer and basketball, are overrepresented, and some have a low number of examples due to these limitations (e.g., sumo wrestling or archery), where not included in analysis. Such imbalance led to the better performance of the model type on the more common categories. Additional work in the future might focus on balancing the dataset by capturing more images from underrepresented classes or using methods like oversampling, SMOTE, etc., to generate synthetic data representing these categories.

Architecture improvements can also depend on the model in question. Although the current CNN worked well among categories, generalization can be substantially improved by incorporating pretrained models, as simplified examples with transfer learning using ResNet, Inception, or Xception, specifically in cases of fewer training images. Pretrained models enable the usage of features learned on large-scale image datasets and transfer them to the sports classification task.

8.2 Dealing With Missing Data and Scaling of the Model

The dataset used for training was relatively clean, but in a real-world setting, there might be instances where data is not available or could be partially populated. In fact, if one wants to use the model on a large scale or in practice, then one should also think about how to handle data missingness and/or corruption (for example, using imputation techniques) — it is very application-dependent.

Then, the next step will be extending this model to classify more than 100 categories, which could be another area of future work. Because the sports world encompasses many different sports and is always changing, making this specific model generalizable so that it can be applied to more sports or other related tasks (for example, action recognition within a sport) would greatly benefit its existence.

8.3 Ethical and Social Implications

Just like for any AI system, it is important to reflect on the ethical implications in deploying this model. On the other hand, we must be careful not to introduce bias by overrepresenting some sports too much and other sports too little in the model. When working to ensure that all sports carry the same weight in media or analytics platforms, special care should be placed on how each sport will be treated equally, rather than just relying on assumed biases.

8.4 Conclusion

In general, this project serves as a starting point for furthering ideas about automated sports classification using deep learning models. This model is capable of providing a lot of value to sports media and analytics , complex models, and deployment strategies are addressed. Future work will concentrate on enhancing the resiliency, scalability, and real-time deployment capacities of our architecture.

References

[1] A. Howard et al., "MobileNets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.

[2] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436-444, 2015.

[3] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *Proceedings of the 32nd International Conference on Machine Learning (ICML)*, 2015.

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

[5] A. Krizhevsky, I. Sutskever, and G. Hinton, "ImageNet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems*, vol. 25, 2012.

[6] D. Scherer, A. Müller, and S. Behnke, "Evaluation of pooling operations in convolutional architectures for object recognition," *Artificial Neural Networks – ICANN 2010*, Springer, 2010.

[7] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929-1958, 2014.

[8] M. Lin, Q. Chen, and S. Yan, "Network in network," *arXiv preprint arXiv:1312.4400*, 2013.

[9] V. Nair and G. E. Hinton, "Rectified linear units improve restricted Boltzmann machines," *Proceedings of the 27th International Conference on Machine Learning (ICML)*, 2010.

[10] C. M. Bishop, *Pattern Recognition and Machine Learning*, Springer, 2006.

[11] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[12] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, MIT Press, 2016.

[13] M. Tan and Q. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," *arXiv preprint arXiv:1905.11946*, 2019.

[14] L. Prechelt, "Early stopping – but when?" *Neural Networks: Tricks of the Trade*, Springer, pp. 55-69, 1998.

[15] G. Piosenka, "Sports Classification Dataset," *Kaggle*, 2022. Available: `https://www.kaggle.com/datasets/gpiosenka/sports-classification`.

[16] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, Springer, 2009.

[17] T. Fawcett, "An introduction to ROC analysis," *Pattern Recognition Letters*, vol. 27, no. 8, pp. 861-874, 2006.

A Model Output Tables

In this section, we present the output of the model predictions for the first 15 samples across selected sports categories.

A.1 Prediction Probabilities for Selected Sports Categories

The following table shows the prediction probabilities for several categories, including *air hockey*, *amputee football*, *archery*, *arm wrestling*, and *axe throwing*.

Sample	**Air Hockey**	**Amputee Football**	**Archery**	**Arm Wrestling**	**Axe Throwing**
0	0.0126	0.0086	0.0067	0.0169	0.0060
1	0.0047	0.0076	0.0050	0.0172	0.0057
2	0.0109	0.0119	0.0098	0.0097	0.0046
3	0.0054	0.0083	0.0083	0.0179	0.0071
4	0.0093	0.0071	0.0045	0.0092	0.0071
5	0.0139	0.0119	0.0068	0.0167	0.0075
6	0.0153	0.0081	0.0034	0.0141	0.0037
7	0.0045	0.0070	0.0085	0.0182	0.0096
8	0.0056	0.0082	0.0051	0.0136	0.0097
9	0.0110	0.0055	0.0075	0.0162	0.0089
10	0.0052	0.0091	0.0103	0.0159	0.0066
11	0.0113	0.0079	0.0059	0.0145	0.0051
12	0.0066	0.0079	0.0060	0.0086	0.0030
13	0.0072	0.0081	0.0075	0.0157	0.0062
14	0.0082	0.0096	0.0079	0.0146	0.0059

Table 1: Prediction probabilities for selected sports categories (first 15 samples)

A.2 Prediction Probabilities for Additional Categories

Similarly, the next table shows probabilities for sports such as *balance beam*, *barell racing*, *baseball*, *basketball*, and *baton twirling*.

A.3 True Labels of the First 15 Samples

The following table shows the true labels of the first 15 samples across different categories.

Sample	Balance Beam	Barell Racing	Baseball	Basketball	Baton Twirling
0	0.0050	0.0065	0.0211	0.0075	0.0087
1	0.0047	0.0055	0.0153	0.0140	0.0209
2	0.0026	0.0071	0.0132	0.0056	0.0187
3	0.0153	0.0122	0.0116	0.0166	0.0166
4	0.0046	0.0052	0.0088	0.0051	0.0144
5	0.0078	0.0101	0.0120	0.0063	0.0146
6	0.0062	0.0090	0.0113	0.0085	0.0181
7	0.0082	0.0086	0.0194	0.0085	0.0114
8	0.0062	0.0107	0.0156	0.0073	0.0182
9	0.0055	0.0072	0.0194	0.0157	0.0172
10	0.0055	0.0063	0.0170	0.0083	0.0157
11	0.0231	0.0102	0.0126	0.0120	0.0166
12	0.0057	0.0066	0.0141	0.0058	0.0140
13	0.0024	0.0045	0.0145	0.0077	0.0123
14	0.0077	0.0068	0.0118	0.0038	0.0160

Table 2: Prediction probabilities for additional sports categories (first 15 samples)

Sample	True Label
0	Air Hockey
1	Air Hockey
2	Air Hockey
3	Air Hockey
4	Air Hockey
5	Amputee Football
6	Amputee Football
7	Amputee Football
8	Amputee Football
9	Amputee Football
10	Archery
11	Archery
12	Archery
13	Archery
14	Arm Wrestling

Table 3: True labels of the first 15 samples