A Hybrid Approach To Stroke Detection In Swimming

A Ankitha Reddy[1,*,†], Pranav Moorthi[2,†], Samyuktaa Sivakumar[3,†], Shwetha S[4,†], Prabavathy Balasundaram[5,†] and Pravinkrishnan K[6,†]

[1]*Sri Sivasubramaniya Nadar College Of Engineering, India*

Abstract

This research explores technology integration in sports, focusing on video-assisted performance diagnostics for swimming. Motion capture techniques, particularly using machine learning models, aim to automate stroke classification, offering efficient analysis of swimmers' techniques. The study proposes a hybrid approach with VGG16 for feature extraction and a Random Forest classifier for stroke classification. Despite challenges stemming from limited data, resulting in an accuracy of 0.28125, the study emphasises the potential of deep learning neural networks for both feature extraction and classification with the aid of larger datasets in the context of swimming performance analysis.

1. Introduction

In recent times, technology has become a crucial part of the world of sports. From being used to scrutinise and analyse athletes' performance during their training period to identifying minor edges that can lead an athlete to victory, cameras and performance monitoring systems are used at every stage of the journey. Applying motion capture techniques through video cameras can go a long way in enhancing a person's capabilities and preventing risks of injuries and player fatigue.

In today's day and age, video-assisted performance diagnostics in the context of swimming have become an indispensable part of improving and enhancing a swimmer's technique. This sort of evaluation of stroke rates, the body postures, and the different phases of a stroke cycle are of great importance to athletes to help them understand how to make movements with minimum movement economy and maximum speed. But it is still largely done manually by replaying and physically noting important features from the video playback. This is not only labour-intensive but also exhausting and time-consuming. Automating this process using state-of-the-art models will largely help in catering to swimmers and athletes who do not have the skill and knowledge to evaluate their technique and are largely dependent on skilled experts for the same.

Using machine learning models to detect the type of swimming stroke is an efficient way to analyse sports performance and derive conclusions about the improvements required in terms of technique and posture. The task [1] given to us is to classify an image into different swimming styles: Freestyle, Backstroke, Breaststroke, Butterfly. All four strokes have their specific technique, and movement economy and have different requirements.

MediaEval'23: Multimedia Evaluation Workshop, February 1–2, 2024, Amsterdam, The Netherlands and Online

*Corresponding author.

†These authors contributed equally.

ankithareddy2210178@ssn.edu.in (A. A. Reddy); pranav221076@ssn.edu.in (P. Moorthi); samyuktaa2210189@ssn.edu.in (S. Sivakumar); shwetha2210210@ssn.edu.in (S. S); prabavathyb@ssn.edu.in (P. Balasundaram)

CEUR Workshop Proceedings (CEUR-WS.org)

2. Related Work

In a study by Hosseini Fani [2], swimming strokes were analysed to predict arm stroke efficiency in videos utilising the OpenPose python library to extract joint features and angles. The classification task utilized the Random Forest technique, achieving an accuracy of 67%.

In another study on recognising basketball turning and dribbling, Zhang et al. [3] introduced flow images to capture the relationship between basketball motions. A convolutional neural network model was utilised with multi-feature learning to extract spatiotemporal features for basketball turning and dribbling recognition effectively.

Furthermore, a study [4] on basketball shooting action efficiency employed a Sparse Gaussian Process Latent Variable Model for motion tracking. Classification methods included Random Forest, Support Vector Machine, SOM neural network, and Bayesian network.

A study [5] addressing stroke detection in tennis videos employed particle filters, motion descriptors and event detectors. The process involved player tracking, extraction of player-centred images, and the use of the Lucas Kanade algorithm for optical flow analysis. Motion descriptors are generated, and feature detection is performed using a 3-D extension of the Viola Jones algorithm. Training utilized the Adaptive Boosting algorithm in machine learning.

Another study [6] on detecting football player activities used CNN and GCN to decipher spatial and temporal patterns in player poses and motions and classify them based on both visual appearances and pose configurations. The model is enhanced with data augmentation and regularisation. The model achieved an F1 score of 0.90.

3. Dataset

The dataset comprised 96 labeled images which were partitioned into an 80 per cent training set and a 20 per cent validation set. The images belonged to 4 classes - backstroke, freestyle, breaststroke, and butterfly, as shown in figures 1-4, with the class-wise distribution being 24.

Figure 1: Freestyle

Figure 2: Breaststroke

Figure 3: Backstroke

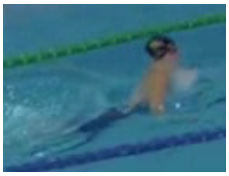

Figure 4: Butterfly

4. Methodology

4.1. Convolutional Neural Networks

Convolutional Neural Network is a widely used machine learning model and is a fundamental element of deep learning algorithms.

A CNN consists of four components: A convolutional layer, an Activation operation, a Pooling layer and a Fully Connected layer. The convolutional layer is the heart of the neural network and is responsible for feature extraction. It involves applying a filter to the image to identify and locate significant features. An element-wise multiplication is performed between the two-dimensional array of weights called the filter and the two-dimensional input array and the result is summed up. This computation is performed using a sliding window [7].

The second component of the CNN is the activation operation. This component is responsible for recognizing specific features in the images it is being trained on and allows the network to learn the non-linear relationships between the weighted sum of the inputs and the output. The most widely used activation functions are Rectified linear unit (ReLU), Sigmoid and Hyperbolic tangent functions.

The third part comprises the pooling layer that is used to reduce the size of the feature map which leads to dimensionality reduction. This sort of down sampling helps in maintaining a lower-resolution feature map while retaining the important features that are important to the classification task. There are largely two types of pooling: Average pooling and Maximum Pooling.

The final component is the Fully Connected(FC) layer. The FC is a densely connected layer whose weights and biases are learned during the training process. This layer's functionality is to flatten the 2D feature map obtained from the previous layers into a 1D array and output a score for each one of the classification classes.

4.2. VGG16

VGG 16 represents a convolutional neural network designed for image classification tasks. Pre-trained on the extensive ImageNet database, this model serves as a feature extractor, demonstrating its proficiency in learning hierarchical representations of visual features. Comprising a total of 16 layers, VGG 16 consists of 13 convolutional layers followed by 3 fully connected layers. The model operates by constructing a 3D tensor, which is subsequently flattened into a 2D vector before being fed into the Random Forest Classifier. The utilisation of a pre-trained model facilitates the extraction of intricate and abstract features.

4.2.1. Random Forest

The Random Forest algorithm serves as an ensemble classification method that leverages the collective decision-making capabilities of multiple decision trees. This method involves the aggregation of individual decision trees to collectively determine the class output. The fundamental unit of the Random Forest is the decision tree, and each tree is trained independently on a randomly selected subset of the training data. An advanced form of the bagging algorithm is employed, introducing an element of randomness to enhance the diversity of the individual trees within the ensemble.

5. Implementation

5.1. Convolutional Neural Networks

The network architecture consists of subsequent Conv2D and MaxPooling2D layers, with the Conv2D layers employing 32 filters of size 3x3 to learn the hierarchical representations. This layer is followed by a MaxPooling2D layer with a 2x2 pooling size, providing a degree of translational invariance. The third Conv2D layer introduces 64 filters, succeeded by a MaxPooling2D layer with a 2x2 pooling size, resulting in a down-sampled output that is reshaped into a one-dimensional array. The architecture includes two dense layers, helping the model understand complex combinations of lower-level features related to different strokes. To enhance regularisation, a Dropout layer with a dropout rate of 0.24 is applied. The architecture employs Rectified Linear Unit (ReLU) activations after each convolutional layer ensuring non-

linearity and spatial down-sampling. Additionally, Softmax activation in the final layer facilitates the generation of output class scores through probabilistic distributions.

5.1.1. VGG16

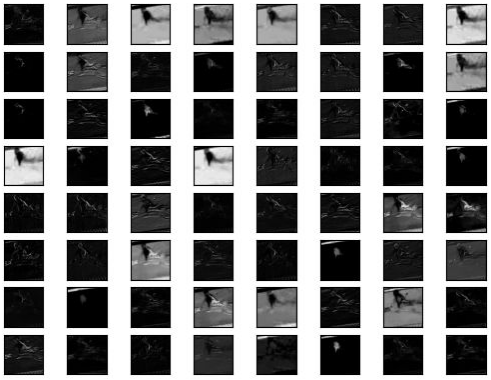

Figure 5: Feature Map pre-VGG16 layer

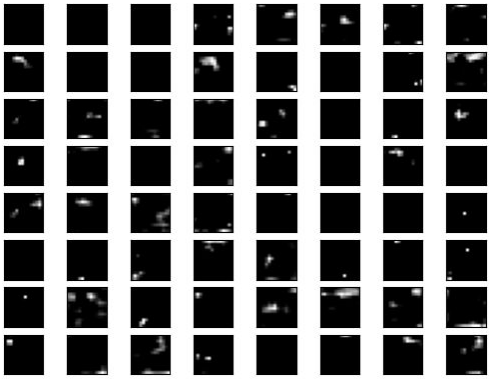

Figure 6: Feature Map post-VGG16 layer

The implementation leverages the VGG16 architecture, consisting of a total of 19 layers with 16 convolutional and 3 fully connected layers. However, only the 16 layers comprising the convolutional base of VGG16 are utilised for the specific purpose of feature extraction. The convolutional layers employ 3x3 filters with a stride of 1, facilitating precise pixel movement. The number of filters increases with the depth of the model, and the architecture is characterised by multiple Conv2D and MaxPooling blocks, each contributing to the extraction of intricate features. Max pooling is executed with 2x2 windows and a stride of 2, efficiently down-sampling the input. Rectified Linear Unit (ReLU) functions are applied to activate the convolutional blocks, providing non-linearity to the model. Addressing the advantages of using smaller filter sizes, hierarchical feature learning and parameter sharing contributes extensively to capturing fine-grained details and reducing the risk of overfitting. The dataset was used to further train VGG16, which was pre-trained incipiently on the extensive ImageNet dataset. Figures 5 and 6 represent the feature maps extracted in each filter before and after the application VGG16 layer.

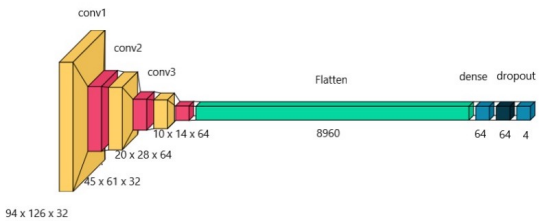

Figure 7: CNN Architecture

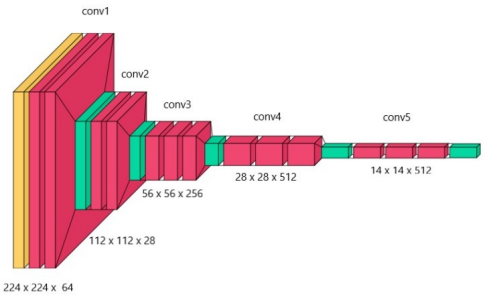

Figure 8: VGG16 Architecture

6. Results and Analysis

The final accuracy of the random forest model on the test data was 0.28125. This low accuracy score is a result of training a neural network like VGG16 on a smaller dataset consisting of 96 images. To accommodate the limited nature of the dataset, we performed data augmentation and opted to use hybrid models that combined deep learning neural networks like VGG16 for feature extraction and traditional models like Random Forest for classification. The CNN model was also applied on the training data which yielded a low validation accuracy of 0.30 due to the extensive dataset requirement of the model. We aimed to extract the most significant features by leveraging state-of-the-art technology and tuning it to fit our dataset by making changes in the hyperparameters, but it underperformed and overfit due to the smaller size of the training dataset.

7. Conclusion

Through the scope of this research, we implemented a hybrid approach. The capability of the CNN and VGG16 models to extract and capture fine-grained details was utilised in identifying the significant features in the images consisting of swimming strokes to correctly classify them.

References

[1] A. Erades, P. Martin, R. V. B. Mansencal, R. Péteri, J. Morlier, S. Duffner, J. Benois-Pineau, Sportsvideo: A multimedia dataset for event and position detection in table tennis and swimming, in: Working Notes Proceedings of the MediaEval 2023 Workshop, Amsterdam, The Netherlands and Online and Online, 1-2 February 2024, CEUR Workshop Proceedings, CEUR-WS.org, 2023.

[2] H. Fani, A. Mirlohi, H. Hosseini, R. Herperst, Swim stroke analytic: Front crawl pulling pose classification, in: 2018 25th IEEE International Conference on Image Processing (ICIP), 2018, pp. 4068–4072. doi:10.1109/ICIP.2018.8451756.

[3] B. Zhang, T. Wang, Visual image recognition of basketball turning and dribbling based on feature extraction., Traitement du Signal 39 (2022).

[4] R. Ji, Research on basketball shooting action based on image feature extraction and machine learning, IEEE Access 8 (2020) 138743–138751. doi:10.1109/ACCESS.2020.3012456.

[5] K. Dokic, T. Mesic, M. Martinovic, Table tennis forehand and backhand stroke recognition based on neural network, in: International Conference on Advances in Computing and Data Sciences, Springer, 2020, pp. 24–35.

[6] L. Lamport, LaTeX User's Guide and Document Reference Manual, Addison-Wesley Publishing Company, Reading, Massachusetts, 1986.

[7] P. Martin, Baseline method for the sport task of mediaeval 2023 3d cnns using attention mechanisms for table tennis stoke detection and classification., in: Working Notes Proceedings of the MediaEval 2023 Workshop, Amsterdam, The Netherlands and Online and Online, 1-2 February 2024, CEUR Workshop Proceedings, CEUR-WS.org, 2023.