# Real-time Pose Recognition for Billiard Player Using Deep Learning

Zhikang Chen, Wei Qi Yan
Auckland University of Technology, 1010 New Zealand

## ABSTRACT

*In this book chapter, we propose a method for player pose recognition in billiards matches by combining keypoint extraction and an optimized Transformer. Given that those human pose analysis methods usually require high labour costs, we explore deep learning methods to achieve real-time, high-precision pose recognition. Firstly, we utilize human key point detection technology to extract the key points of players from real-time videos and generate key points. Then, the key point data is input into Transformer model for pose analysis and recognition. In addition, we design a human skeletal alignment method for comparison with standard poses. The experimental results show that the method performs well in recognizing players' poses in billiards matches and provides real-time and timely feedback on players' pose information. This research project provides a new and efficient tool for training billiard players and opens up new possibilities for applying deep learning in sports analytics. In addition, one of our contributions is the creation of a dataset for pose recognition.*

**Keywords**: Billiards posture analysis, Human skeleton, Key point detection, Transformer, Deep learning

## INTRODUCTION

In recent years, deep learning has been developed rapidly, especially the field of computer vision has become one of the core parts in computing (He et al., 2016). Human pose estimation is one of the essential branches of computer vision, deep learning has proved its effectiveness in dealing with diverse poses, complicated and occlusion problems (Sun et al., 2019). The purpose of human pose estimation is to predict or detect the pose of a human body from an image or video, however, traditional pose estimation methods rely on manual feature input and statistics (Cao et al., 2017). With the emergence of deep neural networks, such as Convolutional Neural Networks (CNN), Transformer and Recurrent Neural Networks (RNN), more accurate and robust methods for human pose estimation were presented (Chen et al., 2018).

Despite significant advances in human pose estimation achieved by using deep learning, many challenges must be addressed, especially in specific areas like billiards. Most existing deep learning methods rely on large-scale labelled data (Wen et al., 2016) and obtain high-quality labelled data in real scenarios which are both time-consuming and laborious. In billiards games, the occlusion of a player's arms, cue, ball, and tabletop makes labelling work even much difficult (Andriluka et al., 2014).

Furthermore, deep learning models perform well on the task of human pose estimation within single images. Still, ensuring temporal continuity and accuracy of human pose estimation in performing continuous actions or video sequences remains a challenge that has yet to be fully addressed (Carreira & Zosserman, 2017). The temporal continuity and accuracy of pose estimation are fundamental in applications such as sports that require high timeliness, e.g., real-time analysis and recommendations of player's poses or stroke strategies (Choutas et al., 2018). In addition, in billiards or other sport games where human poses and environments possess variability and complexity, it may be difficult for a single deep learning model to use all scenarios and actions, which requires solutions with generalization and robustness (Yang et al., 2017).

To address these critical issues, we propose a new approach that combines Transformer and key point recognition to improve the accuracy of billiard player pose recognition while maintaining real-time performance. Meanwhile traditional methods usually ignore timing information, Transformer in deep learning captures this information (Horhreiter & Schmidhuber, 1997) and supports pose estimation with key point information extracted in real time. In order to improve the effectiveness of Transformer, it is necessary to extract the accurate pose features of each image frame. Human key point recognition plays a key role that extracts key point coordinates in real-time and effectively improves the accuracy and real-time performance of player pose recognition.

Applying this deep learning approach is especially important for a sport like billiard games, which requires precise technique and strategy. Our proposed method not only helps specialists to analyse the movements but also provides targeted guidance for players.

Combining Transformer and human key point recognition enables the model to accurately identify key points from single-frame images and analyse player posture changes over time. This research work also has positive implications for billiards training, where players and coaches can utilise these methods to make movement corrections and strategy adjustments to improve skills and performance.

The main contribution of this book chapter is to combine two methods: Transformer and human pose recognition together to address several challenges in billiards player innovatively pose estimation. We also experimentally demonstrate effectiveness and robustness of these methods in various environments, lighting conditions, and player pose analysis.

In this book chapter, we present related work in Section 2, describe our methodology in Section 3 and present our experiments and results in Section 4.

## RELATED WORK

BlazePose is the base model to perform human key point recognition. It targets real-time human pose estimation by segmenting all the essential parts of the human skeleton into 33 key points to ensure the human body's accuracy and robustness of key point recognition (Bazarevsky et al., 2020). In addition, multilevel CNN architecture for key point recognition helps the model to capture finer-grained features and improve recognition accuracy. Meanwhile, combining

depth and spatial pyramid pooling methods, the model can perform feature extraction at different scales, which makes the predicted key points highly satisfactory in all angles and scenes (He et al., 2015). In the face of partial occlusion, the key point extraction algorithm remains robust through the depth structure of the proposed model. It can predict the occluded key points by approximating the contextual information, provided valuable data for real-time human pose recognition (Pauzi et al., 2021). Based on key point recognition, the framework is also employed in human pose tracking (Singh et al., 2021), fitness coaching (Agarwal et al., 2022), animation generation, and augmented reality (Lugaresi et al., 2019). Especially in the fitness domain, by identifying the user's key points, generating a skeletal model and comparing it with a standard posture model, corrective suggestions can be made in real-time to help the user complete the workout while avoiding injuries.

Vaswani et al. introduced the Transformer in 2017, which marked a area from previously dominant architectures based on convolutional neural networks (CNNs) and recurrent neural networks (RNNs) (Vaswani et al., 2017). Distinctively, the Transformer architecture eschews temporal recursion and convolutions, while leveraging instead a self-attention mechanism to enhance performance in sequence-to-sequence learning tasks. This design alleviates the gradient vanishing problem often encountered in RNNs, attributed to the absence of recursive connections in its architecture.

Transformer employs a structure typical of Seq2Seq tasks, comprising an encoder and a decoder (Liu et al., 2018). In Transformer, the encoder and decoder are usually multilayered and ultimately output probabilistic results via softmax. The self-attention mechanism, a core component of the Transformer model, enables the assignment of distinct weights to each unit in the input sequence and recombines these inputs to form a new vector. This mechanism enables the Transformer to discern long-term dependencies within the sequence. The initial phase in computing self-attention entails generating three vectors, namely Query, Key, and Value, for each word vector in the input encoder, with each vector corresponding to a weight matrix. The Query, Key, and Value are derived by multiplying their respective word vectors with these matrices.

Transformer was initially designed to solve the problem of natural language processing, and the power of the self-attention mechanism makes it excellent in other domains as well. Transformer is able to solve visual problems. The Visual Transformer (ViT) guides the Transformer to perform image processing tasks. The main idea of this model is to divide the image into fixed-size blocks and treat these blocks as words in a text sequence. Each image region is linearized into a fixed vector representation, by position coding, ViT can recognize the relative position of individual blocks in the image. Furthermore, a few of experiments have demonstrated that Transformer can outperform traditional CNN models in image classification with sufficient data and resources (Dosovitskiy et al., 2020). Mao et al. proposed a regression human posture estimation framework based on Transformer, which treats human posture estimation as a sequential problem utilizing an attention mechanism to focus the model on the most relevant features at the target key points, which exploits the structured relationships between key points thereby improving performance and avoiding the feature misalignment problem of regression-based methods that was demonstrated to be effective based on the

COCO dataset that Transformer significantly improves the state-of-the-art of regression pose estimation (Mao et al., 2021).

Swin Transformer is designed to be employed as a generalized backbone model for computer vision, which is a new visual Transformer. To process visual data more efficiently, Swin Transformer makes use of a hierarchical Transformer structure and a displacement window approach, the model is also effective in resolving differences in the size of visual entities. Swin Transformer has demonstrated excellent performance in high-resolution tasks like image segmentation and object recognition. In addition, the model outperforms previous state-of-the-art models with COCO and ADE20K datasets (Liu et al., 2021).

## METHODOLOGY

In this book chapter, we create a player's pose dataset for billiard games at various depths, angles, and locations to train the pose detection model. Each was split into a video file of a complete action circle. In the data preprocessing, the player's poses in the video were then processed, and the coordinates of shoulder, elbow, hip, and knee landmarks were acquired for each frame, which are the key nodes for judging whether the player's poses are standard or not, a series of spatiotemporal coordinates were generated from the extracted key point data to depict the poses in motion pictures. The process choice is based on our in-depth understanding of movement, the key nodes are critical for recognising effective human poses.

The collection of pose video data needs time. Finally, 668 videos were captured, and the average completion time of each pose circle is 75 frames after key point extraction. We store the key point data of each video in CSV format for model training and testing, of which 80% was employed as the training set and 20% as the test set. As the amount of data was not as much as expected, we divided the poses into only two classes, i.e., standard and non-standard.

In this book chapter, we employ a model inspired by the foundational Transformer architecture. Initially, the model undergoes a convolutional layer, targeting the extraction of local features. Subsequently, max pooling is applied to down-sample and condense these features, enabling the model to assimilate broader contextual nuances. This is followed by a fully connected layer that seamlessly fuses the feature vectors and reshapes them to dimensions appropriate for the Transformer's input.

Before entering the Transformer's encoder-decoder loops, the data is enriched with positional encodings, ensuring the model is attuned to sequential patterns within the data. The final output from the Transformer is passed through another fully connected layer, which generates probabilistic outcomes in conjunction with a softmax activation. The overarching architecture of our Transformer model is detailed in Figure 1. We implemented the entire model by using the PyTorch framework. The Keras framework as a utility was leveraged to represent the model for visualisation purposes.

We preprocessed the input vectors and planned them into a specific format, i.e., number of samples, number of time steps, and feature dimensions. The number of features comes from analysing the extracted key point information. The output of the BlazePose model contains 33

coordinate points of critical parts of human body, each coordinate point is a $(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$ coordinate containing depth information. Then, we can get the number of features needed as $\boldsymbol{3}\times\boldsymbol{33}$. So, in the input of model structure, we have data in the format of $(\boldsymbol{o}, \boldsymbol{75}, \boldsymbol{99})$, where $o$ is the number of uncertain data samples. The number of times step 75 is calculated from each billiard player striking video we collected.

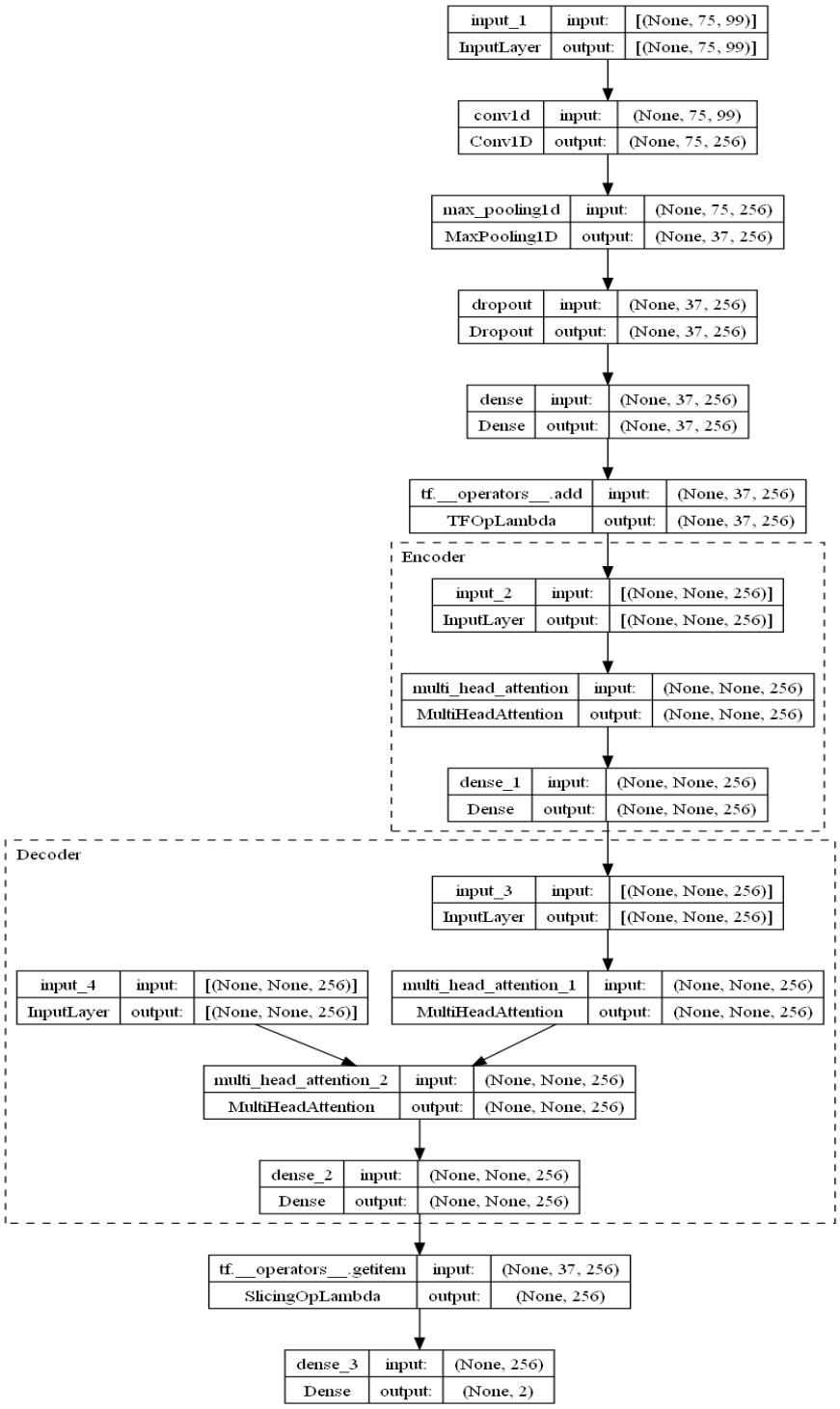



Fig. 1 The structure of Transformer model

The data goes through a one-dimensional convolutional layer before feeding the data into the model. This layer primarily functions to learn the local features of the data. Since the convolution kernel slides through the input sequence, this helps to share the parameters throughout the sequence, thus improving the generalisation ability of this proposed model (Bai et al., 2018). Subsequently, a maximum pooling operation is performed on the data to reduce its dimensionality and improve training efficiency.

Ablation experiments are a classic experimental method in deep learning that explores the importance of each component in a model by eliminating a component in the complete model and evaluating the change in model performance. The experiment helps optimize the model and identify the structure of the top-performing model. Zeiler and Fergus visualized and understood the features of convolutional neural networks through ablation experiments (Zeiler & Fergus., 2014), inverse convolutional networks are employed to map the activation of the network back to the pixel space in order to find out the role of the features. Typically, R-CNN was improved by using ablation experiments to obtain faster real-time response (Sun et al., 2018). In this research project, we show the importance of each component by ablating different components in the whole model and finding the best-performing model structure composition by replacing different methods.

We make use of the key point extraction framework MediaPipe to extract real-time pose bones and compare them with standard pose bones. Scaling and alignment of bones are accomplished by combining algorithms based on a single algorithm. The fusion of algorithmic strategies is chosen based on the effectiveness of the different algorithms for a given task, which ensures fast and accurate comparisons during real-time comparisons.

The follow is a detailed description of the rationale and methodology. The first step calculates the scaling factor. We took use of Euclidean distances for standard and detected key points to calculate the distance between each pair of connections.

$$distance = \sqrt{(x2 - x1)^2 + (y2 - y1)^2} \tag{1}$$

where $(\boldsymbol{x1}, \boldsymbol{y1})$ and $(\boldsymbol{x2}, \boldsymbol{y2})$ are the coordinates of the two key points, and since the current pose comparisons do not involve the computation of depth, we did not use the coordinates $\boldsymbol{z}$ containing depth. Next, the scaling factor is calculated from the average distance between key points.

$$scale_{factor} = \frac{mean(standard_{distance})}{mean(detected_{distance})}. \tag{2}$$

Since the size of detected skeleton displayed on the screen does not match with the size of the standard skeleton, we need to scale the detected bones to match with the standard one. We make use of a scaling factor to calculate the scaled position of the key point coordinates.

$$scale_x = x * scale_{factor} \tag{3}$$

$$scale_y = y * scale_{factor} \tag{4}$$

where $(\boldsymbol{x}, \boldsymbol{y})$ is the original coordinate of the key points. For the scaled key points, we obtain the centre of gravity by calculating the average of their coordinates.

$$centroid_x = \frac{1}{n}\sum_{i=1}^{n} x_i \tag{5}$$

where $\boldsymbol{n}$ is the number of key points and $\boldsymbol{x}$ is the coordinate value, if calculating y coordinate, we need to replace $\boldsymbol{y}$ value with $\boldsymbol{x}$. Finally, we calculate the translation vector and perform the translation. The translation vector is computed based on the centre of gravity of the standard and scaled skeleton.

$$transfer_{vector} = centroid_{standard} - centroid_{scaled} \tag{6}$$

The scaled skeleton will be at a relatively horizontal position to the standard skeleton, and then we take advantage of the computed translation vector to trans-late the real-time skeleton to align with the standard skeleton.

## RESULT ANALYSIS

We firstly conducted ablation experiments on the model to explore the effect of different modules based on the Transformer to optimize the model structure. Then, we conducted controlled experiments with hyperparameters to optimise the parameter settings.

The activation and loss functions are crucial components of the Transformer model, we judge the effect of each function on the model by changing the different functions, we are use of a combination of accuracy, loss values, and F-measure scores for evaluation. Table 1 shows the evaluation scores of the best model after 50 waves of training using different activation functions.

*Table 1 Results of activation function ablation experiments.*

| Activation Function | Accuracy | Loss | F1-Score |
|---|---|---|---|
| None | 98.32% | 0.0957 | 0.9831 |
| Mish | 89.72% | 0.3588 | 0.8971 |
| Swish | 90.28% | 0.4117 | 0.9028 |

The activation function is used after the fully connected layer, the model overall performance without the activation function is better, with an accuracy 98%, a loss value 0.09, and an F1 score 0.98. The model performance is similar when using Mish and Swish. However, the accuracy drops by about 10% compared to the best model performance, and the loss value is much higher than the best performance.

*Table 2 Results of model structure ablation experiments.*

| Model Structure | Accuracy | Loss | F1-Score |
|---|---|---|---|
| Conv and Pooling | 98.32% | 0.0957 | 0.9831 |
| Encoder and Decoder | 83.18% | 0.4541 | 0.8319 |
| Only Encoder | 79.07% | 0.4872 | 0.7903 |

In Table 2, we explore three model structures: A basic Transformer, a Transformer that integrates convolution and pooling operations, and a Transformer that only contains an encoder. The performance of each structure is compared to clarify the impact of the different structures on model performance. Among them, the Transformer with convolution and pooling operations has the best performance, which improves the model accuracy over the base structure by 15% and 19% over the accuracy of the Encoder-only model, it has an average training time 0.7 seconds and an average validation time of less than 0.1 seconds. The encoder model is less accurate but relatively lightweight, with an average training speed of 0.5 seconds and a validation speed of 0.02 seconds.

*Table 4.3 The results of the multiple attention heads ablation experiment.*

| Multi Attention Heads | Accuracy | Loss | F1-Score |
|---|---|---|---|
| 8 | 98.32% | 0.0957 | 0.9831 |
| 4 | 90.09% | 0.3382 | 0.9006 |
| 16 | 90.28% | 0.4336 | 0.9041 |

*Table 4.4 Model dimension ablation experiment results.*

| Hidden Size | Accuracy | Loss | F1-Score |
|---|---|---|---|
| 256 | 98.32% | 0.0957 | 0.9831 |
| 96 | 75.51% | 0.6363 | 0.7550 |
| 512 | 97.20% | 0.1646 | 0.9720 |

*Table 4.5 Results of Num layers ablation experiments.*

| Layer Size | Accuracy | Loss | F1-Score |
|---|---|---|---|
| 2 | 98.32% | 0.0957 | 0.9831 |
| 6 | 88.79% | 0.4355 | 0.8878 |
| 10 | 57.01% | 0.6926 | 0.5530 |

Table 3, Table 4, and Table 5 show the results of the hyperparametric experiments, where the model dimensionality has the most significant impact on performance, with a 23% improvement in accuracy for dimension 256 compared to dimension 96. The increase in the number of layers of the self-attention mechanism also has a significant effect on performance, with the model accuracy after 50 waves of training reaching only 57% if the number of layers is increased to 10 in our batting pose dataset, which may be due to the dataset number being too small to accommodate the self-attention mechanism with more layers.

After completing the ablation experiments, we started training the optimized model. The evaluation criteria of the model are accuracy, loss value and F1 score. In addition, the scenario of the model is real time, so we evaluated response speed. We collected the statistics of the

curves for each evaluation criterion after the training of the model lasted for 50 waves. Figure 2 illustrates the accuracy, loss value and F1 score curve statistics at the end of the training.


*Fig 2 Transformer model accuracy, loss, and F1 score plots.*

We also performed graphical statistics of the model response rate, as shown in Figure 3, where the blue line is the training response rate, which averages around 0.7 seconds, representing the time it takes for the model to receive new data, propagate it, and update the parameters. The red line is the test response speed, which has an average speed 0.02 seconds, ensuring that the model can provide almost real-time feedback when making predictions.

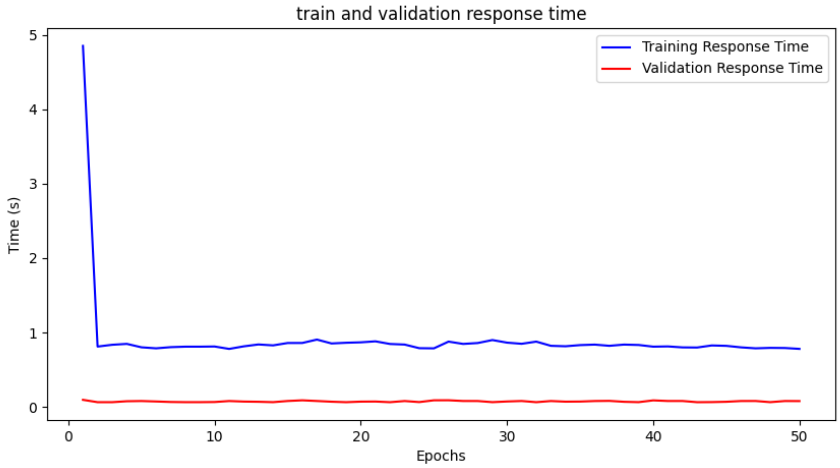



*Fig 3 Transformer training and validation response speed.*

Figure 4 compares the real-time detected bones and the standard pose bones. This feature scales and aligns the real-time detected key points with the standard key point template. We have obtained a standard skeletal model and a real-time detected skeletal model. Then, we combine the two to obtain a real-time pose skeletal comparison system. We scale and align the two models so that the two skeletal models overlap in a predefined area. A zooming algorithm scales the skeletal models to make the two models the same size and keep them on the same level, and then the scaled skeletal models are aligned and overlapped using an alignment algorithm.

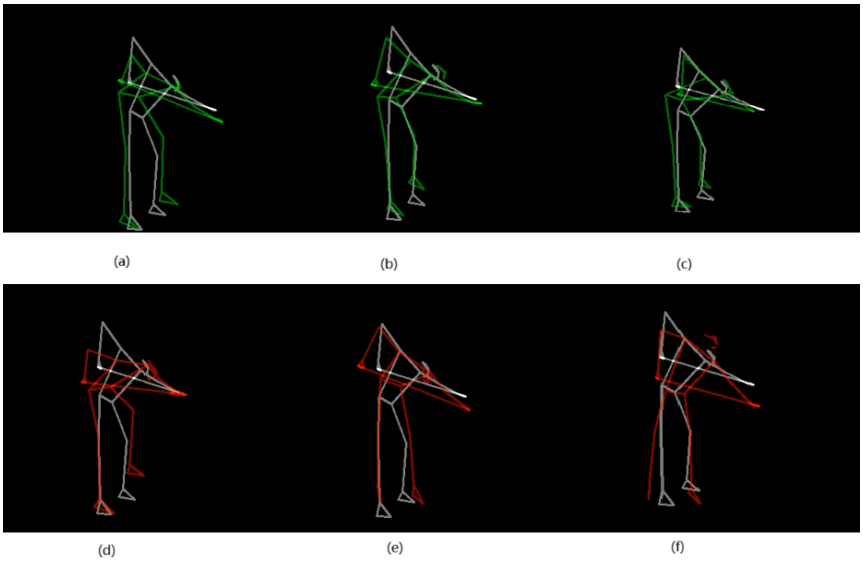
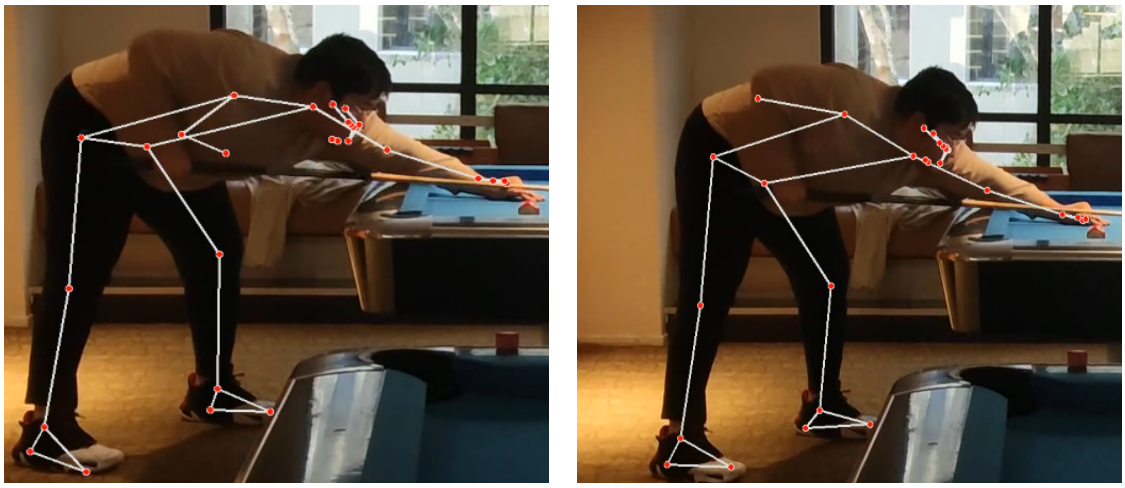



*Fig. 4 The comparison of skeleton detected in real-time with standard skeleton, white for the standard skeleton, green for (a), (b), (c) more standard, and red for (d), (e), (f) less standard.*

We deal with key point fluctuations occurring in the real-time scene through a sliding window and perform the detection of abnormal key point data by determining whether each point in the sliding window exceeds the thresholds of mean and standard deviation. This approach is efficient when dealing with real-time data, the output attitude model can be more stable. Figure 5 shows the effect of key point extraction before and after using the sliding window.



(a) (b)

*Fig 5 Sliding window stabilization pose model*

In the entire system, we firstly extract real-time pose key points and then take advantage of smoothing windows to deal with abnormal key point noise. These extracted key point data are simultaneously computed and analysed in three parts. The first part is the real-time comparison of the skeleton of the key points, which draws the real-time skeleton so that the player can

compare themselves with the standard pose. Secondly, we feed the key point data into the trained Transformer model for real-time prediction, the model will display the output score to the player by calculating the confidence level through the softmax function. This role is to adopt deep learning methods for judging the player's batting posture of the standard and substandard. When the player meets all the evaluation criteria, the system will treat the current action as a standard action, the player can observe through the skeletal comparison of the current action and the standard action that there is still a subtle gap through the posture scores as well as the results of the judgment to provide a more detailed analysis of the defects of the current posture. Figure 6 shows the performance of our proposed method.

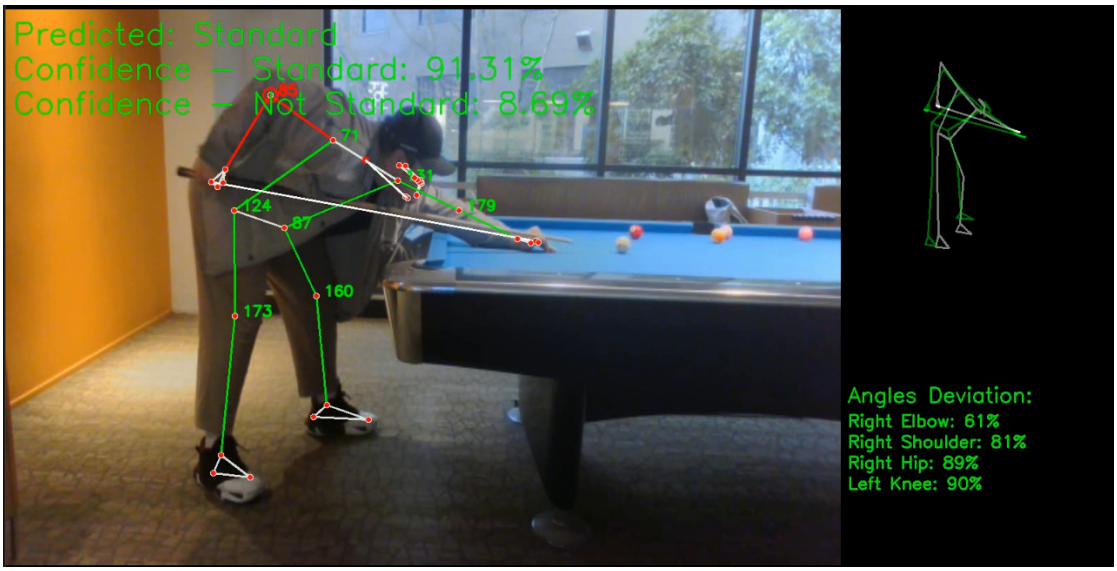



*Fig 6 A system for analyzing the striking posture of billiard players.*

## CONCLUSION

This book chapter aims to combine human pose estimation and realize the striking pose and analysis of a billiard player in a real-time scene. We combine the Transformer with human pose estimation to create a corpus of human poses using key point extraction, which provides a new perspective for the Transformer to recognise and analyse human poses. The extracted key points can also be employed for pose comparison and evaluation. The Transformer model is harnessed to make much accurate and objective judgments on the recognized poses. We also created a dataset for the recognition of striking poses of billiard players, in order to optimize the performance of our model, finally the optimized Transformer achieves an accuracy 98% and a response time 0.02 seconds.

## REFERENCES

An, N., Yan, W. (2021) Multitarget tracking using Siamese neural networks. ACM Transactions on Multimedia Computing, Communications and Applications.

An, N. (2020) Anomalies Detection and Tracking Using Siamese Neural Networks. Master's Thesis. Auckland University of Technology, New Zealand.

Agarwal, V., Sharma, K., & Rajpoot, A. K. (2022), AI based Yoga trainer-simplifying home Yoga using MediaPipe and video streaming. International Conference for Emerging Technology.

Andriluka, M., Pishchulin, L., Gehler, P., & Schiele, B. (2014), 2D human pose estimation: New benchmark and state of the art analysis. IEEE Conference on Computer Vision and Pattern Recognition, 3686-3693.

Bai, S., Kolter, J. Z., & Koltun, V. (2018). An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. arXiv:1803.01271.

Bazarevsky, V., Grishchenko, I., Raveendran, K., Zhu, T., Zhang, F., & Grundmann, M. (2020), Blazepose: On-device real-time body pose tracking. arXiv:2006.10204.

Cao, X., Wei Qi Yan (2023) Pose estimation for swimmers in video surveillance. Multimedia Tools and Applications

Cao, Z., Simon, T., Wei, S. E., & Sheikh, Y. (2017), Realtime multiperson 2D pose estimation using part affinity fields. IEEE Conference on Computer Vision and Pattern Recognition, 7291-7299.

Carreira, J., & Zisserman, A. (2017), Quo vadis, action recognition? A new model and the kinetics dataset. IEEE Conference on Computer Vision and Pattern Recognition, 6299-6308.

Chen, Y., Wang, Z., Peng, Y., Zhang, Z., Yu, G., & Sun, J. (2018), Cascaded pyramid network for multi-person pose estimation. IEEE Conference on Computer Vision and Pattern Recognition, 7103-7112.

Choutas, V., Weinzaepfel, P., Revaud, J., & Schmid, C. (2018), Potion: Pose motion representation for action recognition. IEEE Conference on Computer Vision and Pattern Recognition, 7024-7033.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., ... & Houlsby, N. (2020). An image is worth 16×16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929.

He, K., Zhang, X., Ren, S., & Sun, J. (2015), Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, 37(9), 1904-1916.

He, K., Zhang, X., Ren, S., & Sun, J. (2016), Deep residual learning for image recognition. IEEE Conference on Computer Vision and Pattern Recognition, 770-778.

Hochreiter, S., & Schmidhuber, J. (1997), Long short-term memory. Neural Computation, 9(8), 1735-1780.

Liang, C., Lu, J., Yan, W. (2022) Human action recognition from digital videos based on deep learning. ACM ICCCV.

Liu, C., Yan, W. (2020) Gait recognition using deep learning. Handbook of Research on Multimedia Cyber Security (pp.214-226)

Liu, T., Wang, K., Sha, L., Chang, B., & Sui, Z. (2018). Table-to-text generation by structure-aware seq2seq learning. In AAAI Conference on Artificial Intelligence (Vol. 32, No. 1).

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., ... & Guo, B. (2021). Swin transformer: Hierarchical vision transformer using shifted windows. In IEEE/CVF International Conference on Computer Vision (pp. 10012-10022).

Lu, J. (2016) Empirical Approaches for Human Behavior Analytics. Master's Thesis. Auckland University of Technology, New Zealand.

Lu, J., Shen, J., Yan, W., Boris, B. (2017) An empirical study for human behaviors analysis. International Journal of Digital Crime and Forensics 9 (3), 11-17.

Lu, J., Nguyen, M., Yan, W. (2018) Pedestrian detection using deep learning. IEEE AVSS.

Lu, J., Nguyen, M., Yan, W. (2020) Human behavior recognition using deep learning. International Conference on Image and Vision Computing New Zealand.

Lu, J., Nguyen, M., Yan, W. (2020) Comparative evaluations of human behavior recognition using deep learning. Handbook of Research on Multimedia Cyber Security, 176-189.

Lu, J., Nguyen, M., Yan, W. (2021) Sign language recognition from digital videos using deep learning methods. International Symposium on Geometry and Vision.

Lu, J. (2021) Deep Learning Methods for Human Behavior Recognition. PhD Thesis. Auckland University of Technology, New Zealand.

Lugaresi, C., Tang, J., Nash, H., McClanahan, C., Uboweja, E., Hays, M., ... & Grund-mann, M. (2019), MediaPipe: A framework for perceiving and processing reality. Work-shop on Computer Vision for AR/VR, IEEE Computer Vision and Pattern Recognition, Vol. 2019.

Mao, W., Ge, Y., Shen, C., Tian, Z., Wang, X., & Wang, Z. (2021). TFPose: Direct human pose estimation with transformers. arXiv preprint arXiv:2103.15320.

Pauzi, A. S. B., Mohd Nazri, F. B., Sani, S., Bataineh, A. M., Hisyam, M. N., Jaafar, M. H., ... & Mohamed, A. (2021), Movement estimation using MediaPipe blazepose. IVIC, pp. 562-571.

Singh, A. K., Kumbhare, V. A., & Arthi, K. (2021), Real-time human pose detection and recognition using MediaPipe. International Conference on Soft Computing and Signal Processing, pp. 145-154.

Sun, K., Xiao, B., Liu, D., & Wang, J. (2019), Deep high-resolution representation learning for human pose estimation. IEEE Conference on Computer Vision and Pattern Recognition, 5693-5703.

Sun, X., Wu, P., & Hoi, S. C. (2018). Face detection using deep learning: An improved Faster R-CNN approach. Neurocomputing, 299, 42-50.

Wen, Y., Zhang, K., Li, Z., & Qiao, Y. (2016), A discriminative feature learning approach for deep face recognition. In ECCV pp. 499-515.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. Advances in Neural Information Processing Systems, 30.

Yang, Z., Zhang, K., Liang, Y., & Wang, J. (2017), Single image super-resolution with a parameter economic residual-like convolutional neural network. In MultiMedia Modeling, pp. 353-364.

Wang, S., Zhou,S., Yan, W. (2022) An enhanced whale optimisation algorithm for DNA storage encoding. Mathematical Biosciences and Engineering, 19 (12), 14142-14172.

Wang, X., Yan, W. (2019) Human gait recognition based on self-adaptive hidden Markov model. IEEE/ACM Transactions on Biology and Bioinformatics.
Wang, X., Yan, W. (2019) Cross-view gait recognition through ensemble learning. Neural Computing and Applications.
Wang, X., Yan, W. (2019) Gait recognition using multichannel convolutional neural networks. Neural Computing and Applications.
Wang, X., Yan, W. (2019) Multi-perspective gait recognition based on ensemble learning. Springer Neural Computing and Applications.
Wang, X., Yan, W. (2019) Human gait recognition based on frame-by-frame gait energy images and convolutional long short-term memory. International Journal of Neural Systems.
Wang, X., Yan, W. (2020) Non-local gait feature extraction and human identification. Springer Multimedia Tools and Applications.
Wang, X., Yan, W. (2020) Cross-view gait recognition through ensemble learning. Neural Computing and Applications 32 (11), 7275-7287.
Wang, X., Yan, W. (2022) Human identification based on gait manifold. Applied Intelligence.
Wang, Y. (2021) Colorizing Grayscale CT Images of Human Lung Using Deep Learning. Master's Thesis, Auckland University of Technology, New Zealand.
Wang, Y., Yan, W. (2022) Colorising grayscale CT images of human lungs using deep learning methods. Springer Multimedia Tools and Applications.
Yan, W. (2019) Introduction to Intelligent Surveillance: Surveillance Data Capture, Transmission, and Analytics. Springer London.
Yan, W. (2021) Computational Methods for Deep Learning: Theoretic, Practice and Applications. Springer London.
Zeiler, M. D., & Fergus, R. (2014). Visualizing and understanding convolutional networks. In ECCV (pp. 818-833).
Zhu, Y., Yan, W. (2022) Ski fall detection from digital images using deep learning. ACM ICCCV.