RETRACTION NOTICE

Journal of Applied Science and Engineering (JASE)

Reason for Retraction:
The Journal of Applied Science and Engineering (JASE) is retracting this article due to a fundamental breach of the publication agreement. Following the online appearance of the manuscript, the editorial office was notified that the Article Processing Charge (APC) was processed using a flagged and unauthorized credit card.

As the mandatory financial obligations for publication have not been legitimately fulfilled, this constitutes a violation of the journal's terms and conditions. Consequently, the editorial board has reached the decision to retract the paper from the scholarly record to maintain administrative and ethical integrity.

Editorial Office
Journal of Applied Science and Engineering (JASE)

Yoga Injury Risk Prediction Model Based On Generative Adversarial Network

Rui Tang*

Changchun University of Science and Technology Military Sports Department, Chang Chun, Ji Lin, 130022, China

Corresponding author. E-mail: wg014402@163.com

Received: Aug. 29, 2025; Accepted: Oct. 26, 2025

Incorrect execution of yoga postures, especially in unsupervised environments, increases the risk of musculoskeletal injuries. To address this challenge, a Yoga Injury Risk Prediction Model based on a Generative Adversarial Network (GAN) is proposed, named YOGA-GAN. This model aims to predict injury risks associated with incorrect posture execution. YOGA-GAN leverages the generative capabilities of GANs to create synthetic samples of rare or injury-prone yoga postures, enriching the dataset and mitigating class imbalance issues. The dataset consists of annotated images of both correct and incorrect poses. Preprocessing techniques, such as normalization and Gaussian filtering, are applied to enhance data quality. Feature extraction methods like Histogram of Oriented Gradients (HOG) and Scale-Invariant Feature Transform (SIFT) capture critical structural and textural features of each pose. Principal Component Analysis (PCA) is used for dimensionality reduction while retaining essential information. The discriminator in the GAN is adapted to serve a dual role: distinguishing between real and synthetic poses and predicting the injury risk level of each posture. The system is implemented using Python, providing a robust foundation for developing real-time yoga assistance systems aimed at reducing injury risks. Experimental evaluation demonstrates significant improvements in standard performance metrics, achieving accuracy (96.4%), precision (97.1%), recall (98.2%), and F1-score (97.3%) compared to traditional deep learning models. The YOGA-GAN framework presents a step forward in safer yoga practice and injury prevention.

Keywords: Yoga Injury Prediction, Generative Adversarial Network (GAN), Injury Prevention Systems, injury-prone postures, YOGA-GAN

http://dx.doi.org/10.6180/jase.202607_30.030

1. Introduction

Yoga is an ancient practice combining physical poses (asanas), breathing (pranayama), meditation, and moral principles. It promotes balance between mind, body, and spirit and has evolved into various styles, including Iyengar, Vinyasa, Ashtanga, and Hatha yoga [1]. Yoga is used both as a fitness routine and a spiritual practice. Suitable for all ages and fitness levels, it focuses on mobility, alignment, and mindfulness [2]. Yoga benefits medically, mentally, emotionally, and physically. It supports the treatment of conditions like diabetes, hypertension, arthritis, heart disease, and cancer-related fatigue [3]. Yoga is used in corporate wellness programs, sports training, physiotherapy, rehabilitation, and schools to improve concentration and behavior [4].

The rise in yoga's popularity has led to more injury complaints, especially among beginners and unmonitored practitioners [5]. Yoga injury predictive modelling involves injury probability in a specific asana training of biomechanical form, postural strength, joint stress, and personal health indicators [6].

Traditionally, yoga instructors used observation and experience to improve posture and prevent injuries. However, advancements in deep learning (DL) and machine learning (ML) are increasingly transforming the sector [7]. DL models such as Recurrent Neural Networks (RNNs) and Convolutional Neural Networks (CNNs) have performed moderately well in recognizing [8].

A key challenge is the need for large labeled datasets, which are hard to find in yoga due to the uniqueness of each person's anatomy and preferences [9]. Data augmentation is key to overcoming this limitation. In yoga injury prediction, certain improper or injury-prone poses are rare and under-represented in the collected datasets. If there are not enough samples of certain poses, models may fail to learn the patterns associated with injury risk. Augmentation techniques, especially those that increase the number of artificial samples, will help to balance the dataset, improve generalisation, and will enhance the model's ability to identify small misalignments that lead to injury. Predicting injuries from incorrect posture is underexplored due to sensor integration challenges and limited algorithm processing capacity [10]. A YOGA-GAN was recently proposed to provide injury-related probability predictions in similar cases.

Contributions of this research

Dataset The yoga images and videos dataset was collected.

Pre-processing Gaussian noise filtering and Min-max normalization enhanced data quality, ensuring consistent input and minimizing background noise impact on risk evaluation and yoga posture identification.

Feature extraction SIFT and HOG techniques captured key textural and structural features of yoga poses, enhancing posture change detection. PCA selected relevant features, reducing computational complexity while preserving crucial data for accurate injury risk categorization.

Proposed method The YOGA-GAN architecture was introduced to generate rare improper yoga poses, addressing class imbalance and improving injury risk prediction accuracy.

2. Related works

To improve yoga training participation for the general population by detecting yoga training using the multi-sensor data integration approach in [11]. Yoga training injury scores dropped to 0.02 with added sensors, over 5% lower than previous methods. A transfer learning (TL) strategy with convolutional neural networks (CNN) was used to predict yoga poses in realtime, showing the TL model achieved the highest accuracy and second-best execution time [12].

YOGA-GAN capitalizes on a new application of generative adversarial networks to estimate the risk of injury while contrast to conventional GAN-based approaches, and transfer learning techniques like MediaPipe and MoveNet that focus on general posture analysis and estimating pose. While in the case of MediaPipe and MoveNet, pre-trained models for posture estimations and do not implement injury risk prediction based on pose misalignments. YOGA-GAN adopts new methods of using GANs to create rare or incorrect poses and study those poses for classifying imbalances in the performance of the model to predict injury risk. The YOGAGAN approach simultaneously employs a hybrid approach that leverages feature extraction approaches like HOG, SIFT, and PCA, which improves the models ability to capture important structural and textural metrics in the yoga poses to provide a better assessment of risk.

Deep learning (DL)-based techniques designed to estimate the correct position a practitioner performs were suggested in [13] to make the tasks of such professionals easier. The Media Pipe design achieved the highest accuracy. A yoga tracker using computer vision and ML effectively monitored and corrected poses, improving practice levels [14].

The system for identifying proper yoga positions and offering immediate feedback was developed by [15] using ML approaches and computer vision. The yoga asana detection method showed high accuracy, paving the way for further applications. An automated posture identification technique using MediaPipe and MoveNet outperformed traditional methods [16].

The method in [17] uses DL to identify improper poses and offers correction advice. It outperformed traditional techniques in accuracy with lower computational complexity. The SpYMARNet method, detailed in [18], helps individuals with spondylitis practice yoga poses for treatment. The findings show strong potential for aiding spondylitis patients and offer opportunities for further research in physical therapy and wellness.

A pre-trained MoveNet model and TL, presented in [19], identify ten key yoga poses. Users can compare their current practice with the optimal form for correction. In [20], KNN and PoseNet combine guided meditation, fitness monitoring, and real-time position correction. They analyze camera input and offer feedback to improve posture and reduce injury risk.

Deep Yoga, introduced in [21], uses DL models to enhance posture identification accuracy with minimal resources. [22] explored various yoga position classification methods, presenting the LG Deep model, which combined VGG Net, Squeeze Net, and exception with a unique

RCNN. The LG Deep classifier achieved the highest accuracy, outperforming other techniques.

2.1. Problem statement

Yoga training participation could be tracked, and injury scores progressively decreased with the use of multi-sensor data fusion strategies [11]. Although real-time posture forecasting has been enhanced by TL on CNN [12] and DL-based predictions [13], both methods still have issues with individual posture changes and environmental restrictions. SVMs [14] and other computer vision approaches [15] are the examples of traditional ML models that provide position identification and correction; however, their scalability is limited by their high computing resources and need for controlled circumstances. These issues are addressed by the suggested YOGA-GAN, which is used to predict YOGA risk injury.

3. Methodology

A yoga image and video dataset was collected and preprocessed using Gaussian filter and min-max normalization. HOG and SIFT captured pose details, while PCA reduced feature dimensions. The YOGA-GAN method predicted injury risks from improper pose execution. Figure 1 displays the overview of the YOGA-GAN approach.

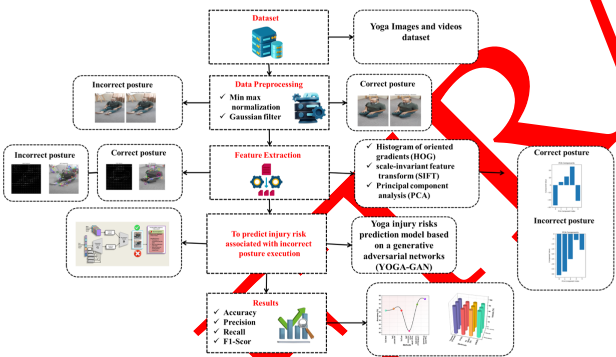

Fig. 1. Overall flow of the YOGA-GAN approach

3.1. Data collection

A dataset of yoga images and videos was collected, including 11,344 compressed images (256×256) in 10 subfolders, each representing a different pose. The video dataset includes 8 videos per pose: 4 showing correct poses from different angles and 4 showing incorrect poses. Figure 2 shows the visualized image distribution per yoga pose. The YOGA-GAN model was trained and evaluated using a dataset that was partitioned into training and testing subsets. The dataset was divided using an $80-20$ split, with the 20% used for testing and the 80% used for training. This split allowed the model to be trained on the appropriate amount of data, while testing its ability to generalize on an unknown set. Another potential option to more rigorously evaluate the model is called K-fold cross-validation. In the K-fold cross-validation approach, the dataset is divisible into K subsets, and the model is trained and evaluated a number of times, each time using a different fold to evaluate.

Source: https://data.mendeley.com/datasets/jc4mmnvcdk/1

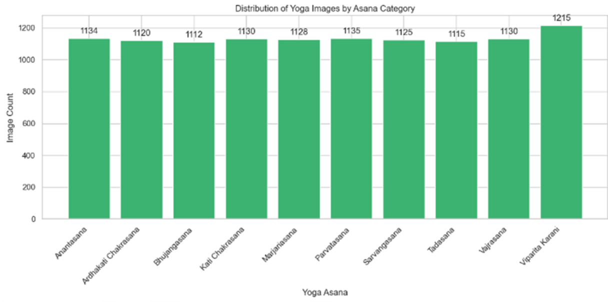

Fig. 2. Image distribution of yoga asanas showing balanced image count per category

3.2. Min-max normalization

Min-max normalization scales injury risk data to the [0,1] range, ensuring feature consistency and improving model efficiency. Image normalization adjusts pixel values to enhance model reliability, strength, and convergence speed.Each pixel intensity is processed using the min-max normalization approach in accordance with Equation (1).

$$w_{\text{norm}} = \frac{w_j - \min(w)}{\max(w) - \min(w)} \tag{1}$$

Where

$\min(w)$ - The image's minimum pixel intensity,

w_j - Image's pixel intensity value, and

$\max(w)$ - The image's maximum pixel intensity.

The min-max normalization is made simpler by Equation (2)

$$w_{\text{norm}} = \frac{w_j}{255} \tag{2}$$

Where

w_{norm} - Normalized pixel intensity.

3.2.1. *Gaussian filtering*

Gaussian filtering smooths video data, removing noise and improving position feature extraction for more reliable injury risk evaluation. It refines images by eliminating dispersed noise before classification. The values of every component in the Gaussian smoothing filter will be created, which can be computed using Equation (3). Figure 3 displays the outcomes of minmax and Gaussian filtered (a) correct posture and (b) incorrect posture images.

$$g(w,z) = \frac{1}{d} e^{\frac{w^2+z^2}{2\sigma^2}} \tag{3}$$

Where d - Normalization constant, σ-Standard deviation (SD) of the Gaussian Kernel.

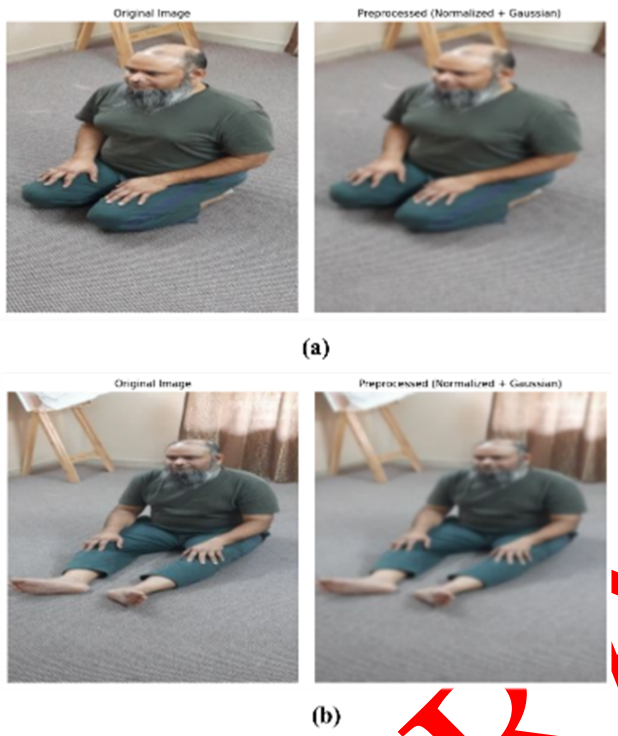

Fig. 3. Outcome of pre-processed images of (a) correct and (b) incorrect yoga poses

3.3. HOG

HOG captures edge and gradient orientation information from yoga posture images, allowing for precise pose evaluation and alignment identification to forecast potential injuries during yoga practice. HOG, in particular, uses convolution kernels to find the horizontal and vertical gradients at each pixel, and then computes the gradient's magnitude and direction. The gradient magnitude and direction are then collected into histograms in cells, which are small spatial divisions. Each histogram represents the distribution of edge directions in that cell. The model compares histograms between cells to discern any small shifts in edge orientations, (e.g., a misaligned knee or kinked spine) that may signal potential bad posture. Because of its cell-wise measurement of gradient patterns, HOG is well-suited for detecting subtle misalignments that may increase injury risk, important information during yoga practice. Equations (4&5) are used to determine the horizontal and vertical gradients of the input image.

$$H_w = J_e * [-1, 0, 1] \tag{4}$$

$$H_z = J_e * [-1, 0, 1]^S \tag{5}$$

The resulting gradients are used with Equations (6&7) to determine the angular orientations and gradient magnitude.

$$n(w,z) = \sqrt{h_u^2(w,z) + h_g^2(w,z)} \tag{6}$$

$$\theta(w,z) = \tan^{-1}\left(\frac{h_u(w,z)}{h_g(w,z)}\right) \tag{7}$$

It separates the image into cells. The attributes are created by combining the blocks, which are made up of different cells. After sorting, these histogram bins are merged to create the final histogram. Subsequently, a description in Equation (8) can be used to calculate the total number of attributes.

$$S_{hog\_fs} = A_{img} * A_t * M_a \tag{8}$$

Where
A_{img} - Blocks per image,
M_a - Number of bins used,
A_t - Block size, and
S_{hog_fs}-Total attributes calculated with the HOG descriptor.

3.3.1. SIFT

SIFT extracts strong key points from yoga pose images to identify injury risks, unaffected by image modifications, rotations, or scaling. SIFT has a multi-step process in order to achieve rotation and scale invariance. In order to find extrema from multiple scales, it creates a scale-space representation using the Difference of Gaussians (DoG). Next, after finding stable extrema and discarding low-contrast or edge-like key points, key points are localized. In order to make sure descriptors are invariant against rotation, a dominant orientation was assigned to each key point based on local gradient directions. Lastly, descriptors were computed for all key points using a histogram of gradients computed within a local region for key points. SIFT is very beneficial for the current injury-risk assessments across different practice environments, owing to this procedure, which allowed it to consistently identify structural elements of yoga poses, even when images varied in size, angle and lighting conditions. It uses DOG to calculate scale extrema and key point localization to remove low-contrast points. Orientation and gradient magnitude are then used to create descriptors for each key point. The following is the SIFT technique,

- Acquiring the Target Image
- Calculating the image's scale space extrema with the DOG function
- Identify the key point of localization
- Calculating the image descriptor.

Each key point's (w, z) estimated local extrema are provided by Equations (9-12),

$$C(w,z,\sigma) = (H(w,z,\sigma) - H(w,z,\sigma)) * J(w,z) \tag{9}$$

$$C(w,z,\sigma) = K(w,z,t\sigma) - K(w,z,\sigma) \tag{10}$$

Where-

$$K(w,z,\sigma) = H(w,z,\sigma) * J(w,z) \tag{11}$$

$$H(w,z,\sigma) = \frac{1}{2\pi\sigma^2} e^{-(w^2+z^2)/2} \tag{12}$$

Where-
$K(w,z,\sigma)$ - Gaussian-smoothed image at the key point σ, and σ-Scaling parameter. Figure 4 displays the output of HOG and SIFT techniques.

The interpolation for every key point is determined using the Taylor series extension of the DOG scale-space operation $C(w,z,\sigma)$, which is provided by Equation (13).

$$C(w) = C + \frac{\lambda C^S}{\lambda w} w + \frac{1}{2} w^S \frac{\lambda^2 C}{\lambda w^2} w \tag{13}$$

The following Equations (14&15) can be used to calculate an image's orientation $\theta(w,z)$ and gradient magnitude $n(w,z)$.

$$n(w,z) = \Big[(K(w+1,z) - K(w-1,z))^2 + (K(w,z+1) - K(w,z-1))^2 \Big]^{1/2} \tag{14}$$

$$\theta(w,z) = \tan^{-1}\left(\frac{K(w,z+1) - K(w,z-1)}{K(w+1,z) - K(w-1,z)}\right) \tag{15}$$

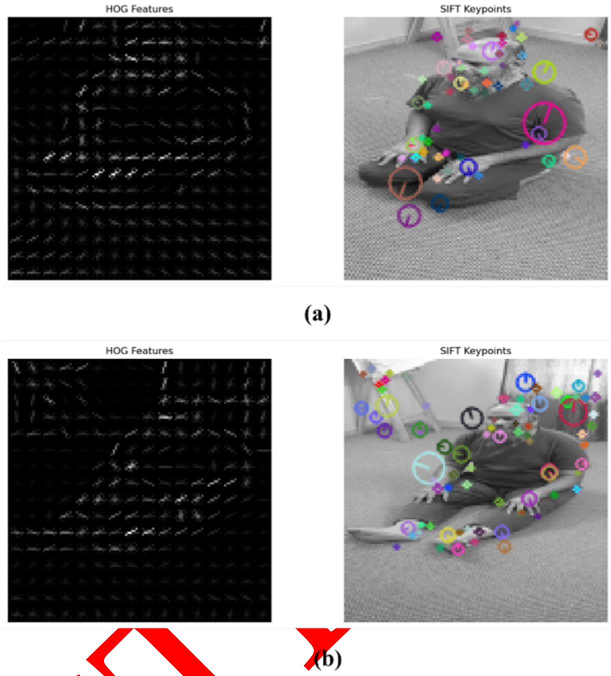

Fig. 4. Outcomes of HOG features and SIFT key points of (a) correct and (b) incorrect

3.3.2. PCA

PCA reduces high-dimensional yoga position data by identifying key features, reducing redundancy, and improving injury risk prediction. It transforms data into a low-dimensional subspace, focusing on the most relevant actions. Each sample in an 1 -sample dataset is mdimensional when ignoring class labels. Consider that $w_1, w_2, \dots\dots\dots w_l \in \Re^m$. The subsequent steps in the PCA computation.
The l-dimensional mean vector μ can be calculated by

$$\mu = \frac{1}{l} \sum_{j=1}^{l} w_j \tag{16}$$

Determine the anticipated covariance matrix T for the collected data by Equation (17),

$$T = \frac{1}{L} \sum_{j=1}^{l} \left(w_j - \mu\right)\left(w_j - \mu\right)^s \tag{17}$$

Determine the associated eigenvectors and eigenvalues of T, where $\lambda_1 \geq \lambda_2, \geq \cdots . \geq \lambda_l \geq 0$.
Determine the l principal components from the l initial variables by Equations (18).

$$\begin{aligned} z_1 &= b_{11}w_1 + b_{12}w_2 + \cdots + b_{1l}w_l, \\ z_2 &= b_{21}w_1 + b_{22}w_2 + \cdots + b_{2l}w_l, \\ z_l &= b_{l1}w_1 + b_{l2}w_2 + \cdots + b_{ll}w_l, \end{aligned} \tag{18}$$

There is no correlation between z_l. The majority of the initial variance in the data set can be explained by z_1, the majority of the remaining variance can be explained by z_2, etc which is shown in Equation (19).

$$\gamma l = \frac{\lambda_1 + \lambda_2 + \cdots\cdots \lambda_n}{\lambda_1 + \lambda_2 + \cdots\cdots + \lambda_n + \cdots + \lambda_l} \geq 80\% \quad (19)$$

Where
γl - The proportion that is maintained in the data representation.

The obtained principal components that can account for at least 80% of the overall variation should be kept when employing PCA for feature extraction. Figure 5 displays the dimensionality feature reduction of (a) correct and (b) incorrect yoga pose images.

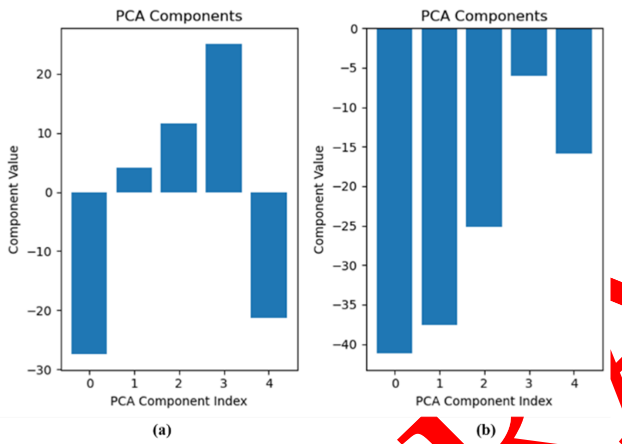

Fig. 5. Dimensionality feature reduction (a) corrects and (b) incorrect yoga pose images

3.4. YOGA-GAN

The YOGA-GAN model predicts injury risks from improper yoga poses by training a discriminator with realistic posture data. It detects misalignments in real-time, helping prevent injuries and promote correct technique during unsupervised or online classes.

The structure of YOGA-GAN is markedly different from a conventional DCGAN in multiple key ways. YOGA-GAN leverages specialized feature extraction methods such as HOG, SIFT, and PCA to capture the structural features of yoga poses - critical information for injury risk assessment - while DCGANs are primarily centered around image generation and simply consist of a generator and discriminator. YOGA-GAN also employs a modified discriminator to assess the poses generated stylistically, in addition to their realism with respect to proper yoga posture, aiming towards improved injury risk assessment. Recent studies have shown the limitations of traditional

DCGANs in generating realistic images and capturing complex relationships and subtle misalignments that are essential for predicting yoga injuries [23]. YOGA-GAN is a new variation on the DCGAN architecture that applies pose-specific features and focuses on predicting injury risk to the precise requirements of yogic practice and injury prevention. The base paper introduces a hybrid framework combining Bayesian inference and Generative Adversarial Networks (GANs) to optimize motion capture systems in physical education. This method addresses key challenges such as data quality, diversity, and uncertainty quantification, improving system accuracy and adaptability. Inspired by this approach, my work extends the application of GANs for synthetic pose generation and injury risk prediction in yoga, leveraging similar AI-driven techniques to enhance injury prevention and personalized feedback in sports science [24].

Model Pruning: This approach reduces the size of the model and speeds up inference by removing some of the weights or parameters of the model, making it useful for devices with restricted resources.

Quantisation: The model size can be significantly reduced and computing efficiency increased by utilising quantisation to reduce the precision of weights (e.g., converting from 32-bit floating point weights to 8-bit integers), particularly on mobile and embedded systems [25].

Edge Computing: It is easy to run real-time injury risk prediction without relying on cloud-based infrastructure by leveraging edge devices, such as smartphones or Internet of Things devices. It can be true of computing configurations due to the ability to process data locally, providing lowlatency processing and better control over your data, leading to enhanced privacy.

Images generated from GANs yield new, realistic poses, and even unusual mistakes that could potentially injure the practitioner, while normal augmentation just changes existing data such as flipping, rotating, or scaling. These artificial postures dramatically increased the model's ability to identify risk of injury and permitted YOGA-GAN to detect small changes in posture while also detecting potentially injurious positions that were unable to be measured when augmentation was without errors. This approach represents a significant development in injury risk assessment. YOGA-GAN directly tackles the class imbalance that often degrades prediction accuracy in yoga datasets by generating synthetic samples of infrequent and injury-risk poses. Unlike standard augmentation methods that only modify existing data (e.g. rotate image), YOGA-GAN creates completely new, realistic representations of incorrect postures. This enriches the training set, and also improves sensitivity and generalization, allowing models to learn from edge cases that are important to prevent injury. What sets YOGA-GAN apart from other GAN-based approaches is its dual capability in injury prediction. In addition to generating realistic yoga poses, the generator aims to generate poses that are uncommon or aberrant, which are often under-sampled in realworld datasets. This alleviates the class imbalance problem which is frequently encountered when training models to predict rare events such as injury detection. Also, YOGA-GAN utilizes HOG and SIFT to extract pose-specific features that are relevant for determining injury risk, while using PCA in dimensionality reduction. In relation to pose estimation systems such as MediaPipe and MoveNet that prioritise pose accuracy, the unique combination of GAN-based image generation and feature extraction methods broadens the model's ability to identify misaligned poses that may lead to injury. Wearable devices or smartphone applications can interact with YOGA-GAN to supply yoga help instantaneously. The user would be engaged with the system through instantaneous input regarding their poses. This can be done through a smartphone application that detects motions and suggests corrections for postures using wearable sensors or the camera on the phone.

For instructors, the system would allow them to monitor poses interacting with multiple students in real-time and notify them when a student is in a misaligned posture which could lead to an injury. This would allow instructors, whether in-person or online, to monitor students while they practice throughout the session and engage in real-time adjustments.

GAN: The model begins with an introduction to the discriminant and generation concepts, followed by an explanation of the GAN optimization functions and how equilibrium between the models is achieved. The generator H takes noise samples y to produce images $H(y)$, while the discriminator C evaluates whether the data is real or generated. C calculates the difference between the two using a distance measure to help H match real data more closely. The model's objective function $U(C,H)$ can be represented as follows in Equation (20),

$$\min_{G}\max_{C} U(C,H) = F_{w\sim o_{\text{data}}(w)}[\log C(w)] + F_{w\sim o_y(y)}[\log(1-C(H(y)))] \tag{20}$$

Where

$H(y)$ - Pseudo-samples

$C(*)$ - Possibility of determining the input sample from the actual sample,

$F(*)$-Distribution function's anticipated value, and

y-Arbitrary noise samples.

When provided H, select C and maximize $U(C,H)$ initially. Then, fix C and minimize $U(C,H)$ to obtain H. When H is provided, maximized $U(C,H)$ calculates the distance or variation between the calculated and actual sample distribution functions. The following Equation (21) must be performed to produce the GAN objective function.

$$\min_{G}\max_{C} U(C,H) = -2\log 2 + 2\,\mathrm{IT\,C}\left(O_{\text{data}}(w)\|o_G(w)\right) \tag{21}$$

Where

$JSD(*)$ - Computation formula for $J-S$ divergence,

$O_{\text{data}}(w)$-The distribution of actual samples,

$o_H(w)$ - Produced sample distribution.

The generator H and discriminator C achieve optimal results when the GAN model converges to specific locations. However, gradient vanishing can occur due to J-S divergence, hindering the desired outcome. Kullback-Leibler (K-L) Divergence, used to calculate J-S divergence, compares two distinct probability distributions, O_1 and O_2, for the same variable w , as shown in Equation (22).

$$C_{LK}(O_1\|O_2) = F_{w\sim O_1}\log\frac{o_1}{o_2} = F_{w\sim O_1}[\log O_1 - \log O_2] \tag{22}$$

The asymmetric characteristic of K-L divergence, $C_{LK}(O_1\|O_2) \neq C_{LK}(O_2\|O_1)$, enables the K-L divergence to have a meaningless value. O_1 and O_2 appear to be inconsistent.

The J-S divergence between two distributions, O_1 and O_2, is defined as the average of the K-L divergence between O_1 and N , and O_2 and N , where $\mathrm{N} = (\mathrm{O}_1+\mathrm{O}_2)/2$, as shown in Equation

$$C_{IT}(O_1\|O_2) = \frac{1}{2}LK\left(O_1\middle\|\frac{O_1+O_2}{2}\right) + \frac{1}{2}LK\left(O_2\middle\|\frac{O_1+O_2}{2}\right) \tag{23}$$

J-S divergence ranges from 0 to log(m), with the maximum occurring when the distributions are far apart. Despite gradient disappearance issues, GAN remains effective across many fields, particularly in image generation. To improve yoga posture synthesis, we incorporated Deep Convolutional (DC) layers into the GAN architecture.

3.4.1. YOGA-GAN

The YOGA-GAN framework builds on the DCGAN design, using transposed convolutional layers in the generator to upsample latent noise vectors into high-quality yoga pose images for injury risk prediction. Multiple architectural choices shape the effectiveness of the DCGAN-inspired

Table 1. Hyperparameters of YOGA-GAN

Category	Hyperparameter	Value / Range
Model Input	Image Size	$128 \times 128 \times 3$
	Latent Dimension (z)	100
Generator (G)	Dense Layer Units	$8 \times 8 \times 256$
	Conv2DTranspose Filters	[128, 64, 32, 3]
	Kernel Size	4×4
	Stride	2
	Activation	Leaky ReLU($\alpha = 0.2$), tanh
	Batch Normalization	After each Conv2DTranspose
Discriminator (D)	Conv2D Filters	[64,128,256]
	Kernel Size	4×4
	Activation	Leaky ReLU($\alpha = 0.2$)
	Risk Output Units	3
	Validity Output Units	1
Loss Functions	Adversarial Loss	Binary Cross entropy
	Auxiliary Loss	Categorical Cross entropy
Optimizer	Type	Adam
	Learning Rate	0.0002
	Beta1	0.5
Training	Epochs	100-300
	Batch Size	32 or 64
	Label Smoothing	0.9 for real, 0.0 fake
	Noise Sampling	Normal (mean $= 0$, std $= 1$)

YOGA-GAN. The generator experiences lay out a latent vector and later upscales the latent vector in an orderly fashion by utilizing Conv2DTranspose along the way to eventually create pose images that respect spatial structure. LeakyReLU activation prevents dead neurons and forwards gradient flow, while batch normalisation is used to stabilize the training by limiting internal covariate shift. All of these characteristics promote convergence in training and shared realism in the images. Furthermore, the discriminator can simultaneously learn the three properties composing pose authenticity and injury risk classification with the auxiliary classifier head, allowing increased stability in predictions and in multimodal learning. Collectively, these scenarios will ensure YOGA-GAN consistently orders safe versus unsafe postures and produces realistic anatomically situated images. The generator projects the latent vector onto a low-quality feature map, then uses Conv2DTranspose layers with batch normalization and LeakyReLU activation to increase spatial resolution. The output layer creates RGB images with Tanh activation. The discriminator, following a standard DCGAN design, downsamples the input image with Conv2D layers and includes an auxiliary classifier head to predict injury risk.Figure 6 displays the structure of the YOGA-GAN approach.

The YOGA-GAN model, based on a DCGAN architecture, generates stable images and predicts injury risk using convolutional and transposed convolutional layers. The generator G(y) maps the high-quality yoga pose image

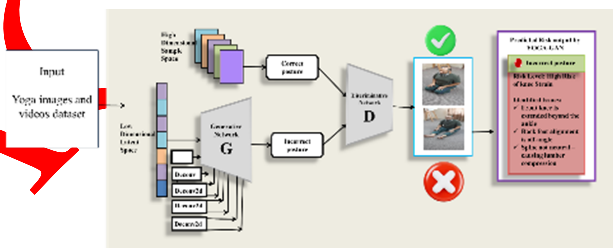

Fig. 6. Architecture of the YOGA-GAN approach for yoga risk prediction

w_correct to a latent noise vector y $\sim$ M(0, J). The equation w_incorrect = G(w) projects w onto a low-quality feature map using stacked Conv2DTranspose layers, LeakyReLU activations, and batch normalization, as shown in Equation (24).

$$g_{j+1} = \text{Leaky ReLU} \left(\text{Batch Norm} \left(\text{Conv2DTranspose} \left(g_j \right) \right) \right) \tag{24}$$

In the final layer, the image output is converted to $[-1, 1]$ using a Tanh activation.

The CNN discriminator C(w) takes an image (real or generated) and outputs two branches: (1) an auxiliary classifier predicting injury risk class C_aux(w), and (2) a real/fake probability C_adv(w) $\in [0, 1]$ indicating authenticity, as shown in Equation (25). The three risk levels of injury low, medium, and high were based on biomechanical criteria and professional annotation. Low risk positions have little to no joint tension and are in correct alignment. Posture from medium-risk poses can cause strain if they were to occur on a persistent basis; however, they would be evident through indications of postural derangements, such as mispositioned limbs or unequal weight distribution. Having high-risk postures increases the potential risk of injury due to misalignment, joint stress, or due to instability as a result of body position. The additional classifier within the discriminator was trained on the aforementioned labels during dataset creation and/or preparation.

$$C(w) = [C_{adv}, C_{aux}(w)] \tag{25}$$

Where-

$$C_{adv} = \sigma\left(X_{adv} \cdot g + a_{adv}\right)$$
$$C_{aux}(w) = \text{softmax}\left(X_{aux} \cdot g + a_{aux}\right)$$

Where, g stands for the down-sampled, compressed feature map that was produced using batch normalization and layered Conv2D layers with LeakyReLU responses, as shown in Equation (26),

$$g_{j+1} = \text{Leaky ReLU}\left(\text{Batch Norm}\ \left(\text{Conv2DTranspose}\left(g_j\right)\right)\right) \tag{26}$$

Auxiliary categorization loss and adversarial loss are combined in the YOGA-GAN loss function, as shown in Equation (27).

$$K_C = K_{adv} + \lambda K_{aux} \tag{27}$$
$$K_G = K_{adv} + \lambda K_{aux}$$

Where-
K_{aux} – Categorical cross-entropy for injury risk forecasting,
λ - Weighing factor that regulates how tasks are balanced, and
K_{adv} - Binary cross-entropy for real/fake discrimination,

YOGA-GAN is an effective model for predicting yoga injury risk, using a DCGAN-based approach. The generator creates realistic yoga images while the discriminator distinguishes real from synthetic images and predicts injury risk. Algorithm shows the Table1 YOGA-GAN pseudocode. Table 1 displays the hyperparameters of the YOGA-GAN approach.

Algorithm 1. YOGA-GAN

Input:

- Yoga Pose Images
- Epochs
- Batch Size
- Latent Dim

Output:

- Trained YOGA-GAN model

Begin:

```
Initialize Generator G and Discriminator D
G generates synthetic yoga poses from latent vector z
D performs two tasks:
    a. Predicts real vs. fake (validity)
    b. Predicts injury risk class (low, medium, high)
Preprocess yoga pose images:
    Normalize pixel values to [−1, 1]
    Apply Gaussian filtering
    Extract HOG/SIFT features (optional)
for epoch = 1 to Epochs do
    for each batch in Yoga Pose Images do
        Sample real images X_real and labels Y_real
        Generate noise vectors Z_noise
        Generate fake images X_fake = G(Z_noise)
        Compute discriminator losses:
    d_real = D(X_real)
    d_fake = D(X_fake)
        d_loss = avg(d_real, d_fake)
        Sample new Z_noise
        Freeze D and train generator
        g_loss = (G+D)(Z_noise)
    Print d_loss, g_loss
    Optionally save generated samples
Save G and D
```

End

4. Results and discussion

The proposed YOGA-GAN approach was evaluated employing the Python platform. The contrast of the methodologies that were used in the current research is accompanied by the important factor of various datasets having been involved in their respective processes, which can, in a way, affect the direct comparability of the results. Future comparisons ought to give emphasis to the datasets that were particularly developed for yoga position detection

Table 2. Results of the ablation study

Configuration	Gaussian filtering	HOG	SIFT	PCA	YOGAGAN	Accuracy (%)	
						Correct Posture	Incorrect Posture
GF only	✓	X	X	X	X	78.2	76.5
HOG only	X	✓	X	X	X	80.1	77.9
SIFT only	X	X	✓	X	X	81.4	78.7
PCA only	X	X	X	✓	X	75.6	73.4
GAN only	X	X	X	X	✓	82.3	80.1
GF + HOG	✓	✓	X	X	X	84.7	82.2
GF + SIFT	✓	X	✓	X	X	85.4	83.0
GF + PCA	✓	X	X	✓	X	83.6	80.8
GF + HOG + GAN	✓	✓	X	X	✓	88.2	86.4
GF + SIFT + GAN	✓	X	✓	X	✓	88.9	87.1
GF + PCA + GAN	✓	X	X	✓	✓	87.5	85.9
GF + HOG + SIFT + PCA + YOGA-GAN (Proposed)	✓	✓	✓	✓	✓	95.4	97.4

and injury risk prediction. The performance of the suggested method was evaluated with traditional methods such as Random Ro-rest (RF) [26], Support Vector Machine (SVM) [26], extreme Gradient Boosting (XG Boost) [26], Yoga Long Short-Term Memory (LSTM) Pose Machines [27], and Long ShortTerm Memory (LSTM)-Based Model [27].

4.1. Accuracy and loss monitoring trends for yoga injury risk prediction

Figure 7 shows the yoga injury risk forecasting model's performance over 50 epochs. Validation accuracy rises from 0.78 to 0.95 , while training accuracy increases from 0.71 to 1.0 , indicating strong learning with a slight generalization gap. Validation loss decreases from 0.55 to 0.12 by epoch 20, while training loss drops rapidly from 0.61 to 0.01 . The continued decrease in training loss and stable validation loss suggests slight overfitting. The model predicts injury risk well but needs adjustments for better validation performance.

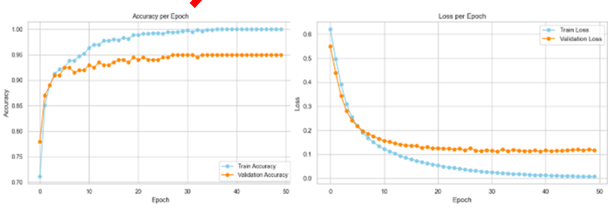

Fig. 7. Training and validation comparison of accuracy and loss in a yoga injury risk assessment model

The YOGA-GAN method analyzes yoga poses to identify misalignments and predict injury risks. It generates realistic pose variants and detects deviations from standards, reducing musculoskeletal injury risks by highlighting joint posture issues. Figure 8 shows examples of correct (left) and incorrect (right) poses detected by YOGA-GAN.

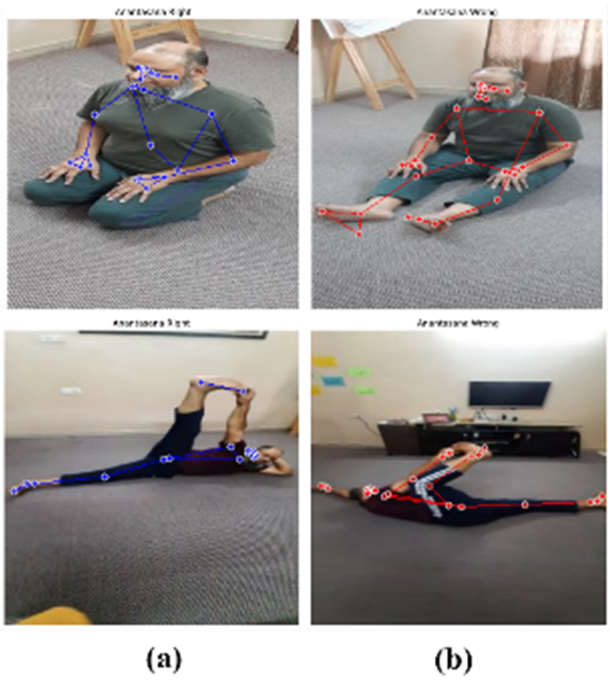

Fig. 8. Injury risk prediction of (a) correct and (b) incorrect yoga posture

Table 3. Comparative analysis of accuracy metrics of existing RF, SVM, XG Boost, Yoga LSTM Pose Machines, and LSTM-Based Model, and the proposed approach

Methods	Accuracy (%)	Dataset Used
SVM [26]	90.78	Yoga Pose Recognition Dataset
RF [26]	90.78	Yoga Pose Recognition Dataset
XG Boost [26]	91.64	Generic Pose Dataset
Yoga LSTM Pose Machines [27]	80.85	Yoga Pose Dataset
LSTM-Based Model [27]	93.75	Yoga Pose Dataset
YOGA-GAN [Proposed]	96.4	Yoga Pose Recognition Dataset

Table 2 shows the ablation study results for correct and incorrect yoga postures. Preprocessing techniques like Gaussian filtering (78.2%, 76.5%), HOG (80.1%, 77.9%), SIFT (81.4%, 78.7%), PCA (75.6%, 73.4%), and GAN (82.3%, 80.1%) achieved moderate accuracy. Combining features, the proposed approach (GF + HOG + SIFT + PCA + GAN) outperformed others, with 95.4% accuracy for incorrect and 97.4% for correct pose predictions.

Performance evaluation of the proposed YOGA-GAN method using various performance indicators

- The accuracy assesses how exactly the system recognizes proper versus incorrect yoga poses, demonstrating its efficiency in reducing injury risks by accurately identifying safe and harmful movements.
- Precision calculates the percentage of accurately anticipated injury risks among all projected injury instances, ensuring accurate recognition and reducing false alarms during posture assessment.
- The recall checks the ability of the model to detect real threats of injury and minimize the cases of false negative and eliminate the possibility of real dangerous postures going undetected each time.
- The F1-score statistic is ideal in determining the performance of prediction in the real practicing environment because it balances away precision and recall, hence ensuring that the model does not miss a risky posture keeping false alarms to the minimum.

The proposed YOGA-GAN strategy for yoga injury risk forecasting was assessed employing several performance metrics. The suggested approach achieved better results in terms of accuracy (96.4%), F1-score (97.3%), precision (97.1%), and recall (98.2%). Performance indicators were enhanced, and the model's ability to identify abnormal postures prone to injury was enhanced with the use of GAN-generated images. Abnormal misalignments and postures could be detected more accurately in estimating injury risk thanks to these artificial images, which depict distinctly abnormal misalignments that are often less represented in traditional datasets.

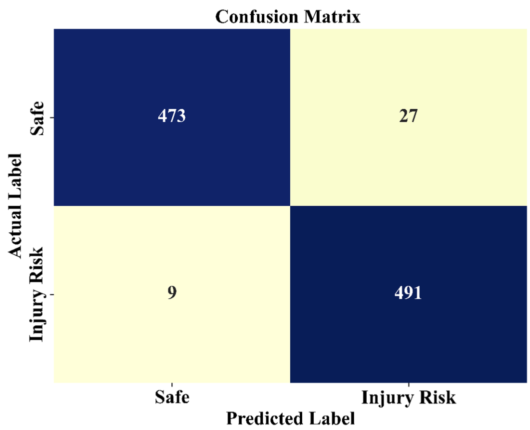

Fig. 9. Confusion matrix showing classification results for "Safe" and "Injury Risk" categories predicted by YOGA-GAN

In order to evaluate classification performance for the binary task of injury risk prediction, a confusion matrix was created. As shown in Figure 9, the model correctly identified 491 injuryrisk poses and 473 safe postures, with few misclassifications. This visualization demonstrates that the model was found to be very reliable in distinguishing between safe and dangerous poses, providing some credence to its real application in real-time yoga support systems.

4.2. Discussion

Yoga, a traditional practice, is widely popular for its benefits to the body and mind. The complexity of motor control and subtle relationships in position shifts makes it difficult to model yoga-related injuries accurately in RF [26]. SVMs [26] also require extensive preprocessing for time-series data, limiting their ability to predict injury risk.

Yoga injury risk forecasting with XGBoost [26] is limited by its inability to detect temporal relationships in posture streams, Future research should ensure that these kinds of comparisons are performed with datasets particularly

Table 4. Performance analysis of existing and proposed methods

Methods	F1-score (%)	Precision (%)	Recall (%)
SVM [26]	90.01	91.78	89.63
RF [26]	90.22	91.07	89.83
XG Boost [26]	91	91.64	90.65
YOGA-GAN [Proposed]	97.3	97.1	98.2

designed for yoga postural analysis and injury risk assessment, to guarantee a fair assessment, even though the methodologies utilized in this comparison were assessed with different datasets. relying heavily on feature extraction. LSTM Pose Machines [27] face overfitting issues with imbalanced datasets and require lengthy, labeled posture sequences, which are hard to collect. LSTM models also struggle with noisy sensor data and small posture imbalances, and their computational complexity limits real-time use in wearable or mobile yoga systems. The YOGA-GAN method outperformed RF, SVM, XGBoost, Yoga LSTM Pose Machines, and LSTM-Based Models with 96.4% accuracy, compared to 90.78%, 90.78%, 91.64%, 80.85%, and 93.75%, respectively (Figure 4). Synthetic Data Generation: Unlike conventional models such as SVM and XGBoost which struggle with uncommon events due to limited data, GANs are capable of producing synthetic data to mitigate class imbalances by generating rare and injury-prone positions. Learning from Raw Data: GANs learn directly from raw data, which allows them to identify complex high-dimensional patterns in poses, improving the effectiveness of predicting injury risk when compared to models such as SVM and Random Forest that require explicit feature extraction. GANs are more skilled at modeling complicated relationships and small misalignments that contribute to injury than simpler models, such as Random Forest or XG-Boost, that rely primarily on features and decision trees.

Improved Performance: YOGA-GAN showed that by combining pose identification with injury risk prediction into a single model was more effective in predicting injury risk, outperforming SVM, XGBoost, and RF in regard to precision, recall, and F1-score as shown in Figure 10 and Tables 3-4. With an F1-score of 97.3%, YOGA-GAN surpassed RF (90.22%), SVM (90.01%), and XGBoost (91%) (Figure 5). Its precision of 97.1% outperformed RF (91.07%), SVM (91.78%), and XGBoost (91.64%) (Figure 6). YOGA-GAN's recall of 98.2% exceeded the low recall values of RF (89.83%), SVM (89.63%), and XGBoost (90.65%) (Figure 10). Tables 3 and 4 show the precision, F1-score, accuracy, and recall for all methods.

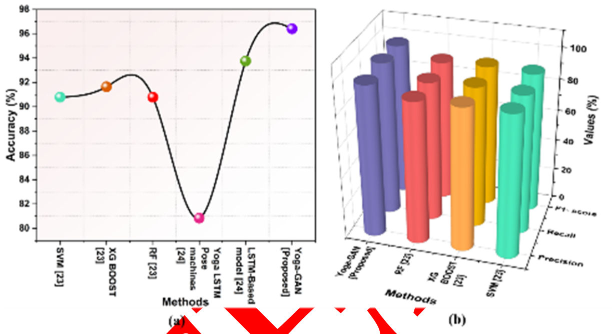

Fig. 10. Comparative assessment of (a) accuracy and (b) recall, F1-score, and precision

5. Conclusion

Yoga, a traditional discipline combining physical postures, breathing, and meditation, has gained global popularity for improving health, focus, and fitness. A dataset of yoga images and videos was collected and preprocessed using Min-max normalization and Gaussian filtering. Key features were extracted using HOG, SIFT, and PCA. The YOGA-GAN method was developed to predict injury risk from incorrect yoga poses, with performance evaluated at 97.1% precision, 97.3% F1 score, 96.4% accuracy, and 98.2% recall. Physiological Differences: By incorporating additional physiological data, including age, strength, flexibility, and body type, the model could be updated. By making decisions based on differences in body type or flexibility levels, this would allow for a more individualized injury risk assessment, as different body types or levels of flexibility may have different risks of injury during specific poses. For example, the model may be updated to consider that older practitioners, or practitioners who are less flexible, may be more likely to injure themselves during specific poses. Differences Based on Skill: The model could potentially be altered to recognize skill levels, as novice practitioners are at a greater risk of injury due to poor pose execution. The model could employ feedback loop based on skill levels where more experience practitioners receive less corrective feedback (with a focus on the advanced errors) and more novice practitioners see adjustments in posture correction. This would allow for a more individualized injury prediction model, increasing the accuracy of the model and the credibility and usability of the model outside of a laboratory context.

Future work should integrate motion capture, biomechanical modelling, and wearable sensor data to improve the model's ability to predict injury. Wearable sensors are capable of capturing force, joint tension, and muscle strain via real-time data capture about internal stresses that a standard postural assessment would not ordinarily be able to capture. In addition, motion capture and biomechanical modelling can provide more comprehensive insight into the kinematics and dynamics of each pose, allowing for a better understanding of injury risk associated with muscular and skeletal stresses that may lead to injury, especially when obvious misalignments may not be adequate to predict injury. Limitations include the lack of diverse datasets, individual posture variation, inability to capture real-time environmental factors, overfitting, limited generalizability, and difficulty in detecting subtle posture variations. Future improvements include integrating wearable sensors, expanding datasets, improving accuracy, developing mobile apps, and incorporating biomechanical analysis for better injury prevention.

Declarations

Fundings: Jilin Provincial Department of Education "13th Five-Year Plan" social science projectResearch on the Construction and Application of Yoga Sports Exercise Mode for College Students (JJKH20200811SK)
Conflicts of statement: Not Applicable.
Data availability: Not Applicable.
Code availability: Not Applicable.
Author's Contributions: Rui Tang is responsible for designing the framework, analysing the performance, validating the results, and writing the article.

References

[1] R. Barua. "Harmonizing Body and Mind: Investigating the Combined Impact of Yoga and Mindfulness Meditation on Depression, Anxiety, Stress". In: *Global Innovations in Physical Education and Health*. IGI Global, 2025, 203–232. DOI: https://doi.org/10.4018/979-8-3693-3952-7.ch008.

[2] M. Pundir, V. S. Yadav, A. R. Roj, S. Kumar, and R. Sharma, (2025) *"Yoga Therapy: A Holistic Approach to Enhancing Physical, Psychological, and Social Health"* **Indian Journal of YOGA Exercise & Sport Science and Physical Education**: 18–30. DOI: https://doi.org/10.58914/ijyesspe.2025-10.1.4.

[3] G. Q. Liu, J. Y. Han, W. J. Song, and Y. Wang, (2022) *"Research Progress of Yoga in Symptom Management of Cancer Patients"* **TMR Non-Drug Therapy** *5*(2): 10. DOI: https://doi.org/10.53388/TMRND20220531010.

[4] R. Bucea-Manea-Țoniș and D. G. Păun, (2024) *"Yoga, a Mindfulness Therapy to Prevent PTSD as to Encompass Athletes' Performance"* **Frontiers in Psychology** *15*: 1334278. DOI: https://doi.org/10.3389/fpsyg.2024.1334278.

[5] G. Wei and L. Yi, (2025) *"Predicting the Impact of Yoga on Chronic Venous Insufficiency: A Machine Learning Approach Using Naive Bayes Classifier and Optimization Systems"* **Journal of Ambient Intelligence and Humanized Computing**: 1–23. DOI: https://doi.org/10.1007/s12652-024-04889-6.

[6] X. Du, (2024) *"Forecasting the Yoga Influence on Chronic Venous Insufficiency: Employing Machine Learning Methods"* **International Journal of Advanced Computer Science & Applications** *15*(3): DOI: https://doi.org/10.14569/ijacsa.2024.0150345.

[7] X. Han and N. Hu, (2024) *"Prediction of One- and Three-Months Yoga Practices Effect on Chronic Venous Insufficiency Based on Machine Learning Classifiers"* **Egyptian Informatics Journal** *27*: 100507. DOI: https://doi.org/10.1016/j.eij.2024.100507.

[8] J. Palanimeera and K. Ponmozhi, (2024) *"Yoga Posture Recognition by Learning Spatial-Temporal Feature with Deep Learning Techniques"* **International Journal of Image and Graphics** *24*(06): 2450055. DOI: https://doi.org/10.1142/S0219467824500554.

[9] D. Swain, S. Satapathy, B. Acharya, M. Shukla, V. C. Gerogiannis, A. Kanavos, and D. Giakovis, (2022) *"Deep Learning Models for Yoga Pose Monitoring"* **Algorithms** *15*(11): 403. DOI: https://doi.org/10.3390/a15110403.

[10] R. Pal, D. Adhikari, M. B. B. Heyat, I. Ullah, and Z. You, (2023) *"Yoga Meets Intelligent Internet of Things: Recent Challenges and Future Directions"* **Bioengineering** *10*(4): 459. DOI: https://doi.org/10.3390/bioengineering10040459.

[11] J. Liu and Y. Li, (2024) *"Yoga Training Injury Detection Method Based on Multi-Sensor Information Fusion"* **Internet Technology Letters** *7*(5): e435. DOI: https://doi.org/10.1002/itl2.435.

[12] N. Maddukuri and S. R. Ummity, (2023) *"Yoga Pose Prediction Using Transfer Learning Based Neural Networks"*: DOI: https://doi.org/10.21203/rs.3.rs-2807080/v1.

[13] D. M. Kishore, S. Bindu, and N. K. Manjunath, (2022) *"Estimation of Yoga Postures Using Machine Learning Techniques"* **International Journal of Yoga** ***15***(2): 137–143. DOI: https://doi.org/10.4103/ijoy.ijoy_97_22.

[14] D. Mane, G. Upadhye, V. Gite, G. Sarwade, G. Kamble, and A. Pawar, (2023) *"Smart Yoga Assistant: SVM-Based Real-Time Pose Detection and Correction System"* **International Journal on Recent and Innovation Trends in Computing and Communication** ***11***(7): 251–262. DOI: https://doi.org/10.17762/ijritcc.v11i7s.6997.

[15] D. Borthakur, A. Paul, D. Kapil, and M. J. Saikia, (2023) *"Yoga Pose Estimation Using Angle-Based Feature Extraction"* **Healthcare** ***11***(24): 3133. DOI: https://doi.org/10.3390/healthcare11243133.

[16] D. Parashar, O. Mishra, K. Sharma, and A. Kukker, (2023) *"Improved Yoga Pose Detection Using Media Pipe and Move Net in a Deep Learning Model"* **Revue d'Intelligence Artificielle** ***37***(5): DOI: https://doi.org/10.18280/ria.370511.

[17] V. Anand Thoutam, A. Srivastava, T. Badal, V. Kumar Mishra, G. R. Sinha, A. Sakalle, H. Bhardwaj, and M. Raj, (2022) *"Yoga Pose Estimation and Feedback Generation Using Deep Learning"* **Computational Intelligence and Neuroscience** ***2022***(1): 4311350. DOI: https://doi.org/10.1155/2022/4311350.

[18] A. Rawat, A. Balasundaram, and C. A. Vaithilingam, (2025) *"Angle-Based Regularized Deep Learning Model for Gauging Effectiveness in Performing Yoga Postures"* **Sport Sciences for Health**: 1–17. DOI: https://doi.org/10.1007/s11332-025-01328-7.

[19] S. Yenishetti, G. Karajkhede, and L. Panat. "Virtual Yoga Teacher with AI Model for Yoga Posture Correction for Good Health". In: *2024 ITU Kaleidoscope: Innovation and Digital Transformation for a Sustainable World (ITU K)*. IEEE, 2024, 1–8. DOI: https://doi.org/10.23919/ITUK62727.2024.10772930.

[20] S. Mishra, I. Bhardwaj, A. Singh, T. Chakraborty, and A. Tripathi. "Real-Time Pose Correction and Wellness Tracking System for Enhanced Yoga Practice". In: *Challenges in Information, Communication and Computing Technology*. CRC Press, 2025, 383–388. DOI: http://dx.doi.org/10.1201/9781003559092-66.

[21] R. Bansal, R. Sharma, P. Jain, R. Arora, and S. Pal, (2024) *"Deep Yoga: Enhancing Practice with a Real-Time Yoga Pose Recognition System"* **Engineering, Technology & Applied Science Research** ***14***(6): 17704–17710. DOI: https://doi.org/10.48084/etasr.8643.

[22] A. S. Talaat, (2023) *"Novel Deep Learning Models for Yoga Pose Estimator"* **SN Applied Sciences** ***5***(12): 341. DOI: https://doi.org/10.1007/s42452-023-05581-8.

[23] H. Jubair, R. T. Chowdhury, F. Akter, I. S. Sami, N. Nawal, S. T. Arshi, and P. Shil, (2025) *"Machine Learning for Real-Time Exercise Correction and Injury Prevention: A Systematic Review"* **Machine Learning** ***13***: 1.

[24] D. K. Reddy Basani, R. Lakshmi Gudivaka, S. H. Grandhi, B. Ramanjaneyulu Gudivaka, R. K. Gudivaka, and M. M. Kamruzzaman. "AI-Powered Motion Analysis in Physical Education: GANs and Bayesian Approaches". In: *2025 4th OPJU International Technology Conference (OTCON) on Smart Computing for Innovation and Advancement in Industry 5.0*. Raigarh, India: IEEE, 2025, 1–6. DOI: https://doi.org/10.1109/OTCON65728.2025.11071118.

[25] A. Wang, X. Wu, L. Zhao, H. Chen, and S. Zhao, (2021) *"Physical Activity Recognition from Accelerometer Data Using Multi-View Aggregation"* **Journal of Applied Science and Engineering** ***24***(4): 30 April 2021. DOI: https://doi.org/10.18280/jase.240416.

[26] A. Jha. *"Leveraging OpenCV for Precise Yoga Pose Estimation and Reducing Injury Risks"*. Doctoral dissertation. (phdthesis). Dublin: National College of Ireland, 2025.

[27] E. Bagga and A. Yang. "Real-Time Posture Monitoring and Risk Assessment for Manual Lifting Tasks Using Media Pipe and LSTM". In: *Proceedings of the 1st International Workshop on Multimedia Computing for Health and Medicine*. IEEE, 2024, 79–85. DOI: https://doi.org/10.1145/3688868.3689199.