

# Modelo para la Detección de Posturas de Yoga Usando Redes Neuronales Convolucionales

Oskar Ramses Beltran Magaña  
*Universidad Veracruzana*  
*Facultad de Ingeniería Eléctrica y Electrónica*  
 Veracruz, Mexico  
 oskarr.magana@gmail.com

**Abstract**—Yoga is a discipline many people want to get involved to due to its benefits on general well being. The new lifestyles derived in the necessity of systems that are able to teach and supervise the execution of yoga postures. This project proposes model that classifies yoga postures using computer vision and convolutional neural networks.

**Index Terms**—yoga, pose, detection, CNN.

## I. INTRODUCCIÓN

ENTRE las distintas causas de lesiones al ejercitarse, podemos encontrar la incorrecta ejecución de las actividades [1]. Por otro lado, para la recuperación en algunas de estas lesiones es necesario realizar otro grupo de ejercicios cuya efectividad viene dada a partir de la adecuada ejecución de estos [2]. Sin embargo, no siempre es posible supervisar que las actividades se realicen apropiadamente, por lo que un sistema que sea capaz de realizar esta tarea permitiría aumentar la eficacia de las rutinas.

La Yoga es una filosofía que, además ideas espirituales y filosofías alimentarias, utiliza distintas posturas muy marcadas que requieren de su correcta ejecución para conseguir diversos efectos en el cuerpo de quien las lleva a cabo. Debido a los efectos que estas posturas pueden tener en el bienestar de quien las hace, es que la gente busca practicarlas sin embargo al no siempre es posible asistir con un maestro yogi que supervise el desarrollo de estos ejercicios.

Por ello, esta investigación tiene como objetivo el desarrollar un modelo que explore la capacidad de identificar entre cinco poses distintas a partir de una clasificación simple.

## II. ESTADO DEL ARTE

En la detección de posturas humanas han existido distintas aproximaciones y el caso de la yoga no es la excepción. Los principales enfoques para para detección de estas han sido dos: una sola persona y múltiples individuos. Así pues, en la línea de una sola persona, Palanimeera y Ponmozhi [3] crearon un algoritmo de dibuja un esqueleto sobre el individuo y comparan la eficacia de cuatro algoritmos diferentes a la hora de clasificar la postura del saludo al sol, siendo el de K vecinos más cercanos el de mayor eficacia con un 96% de precisión. Bhosale et al. [4] proponen el uso de una librería pre-existente para la detección de las posturas y, posteriormente, a través de un algoritmo de K vecinos más

cercanos realizar la clasificación de entre 5 posturas diferentes. Haciendo uso también de los K vecinos más cercanos. Una propuesta interesante es la que se propone en [5] donde el modelo busca no sólo aprender de un set de datos preexistente, si no que toma un video que el usuario agrega para determinar y corregir las posturas de este. Esta propuesta realiza un dibujado de las articulaciones para crear una figura del sujeto y con ella evaluar las posturas a través de cuatro modelos distintos, siendo el de redes convolucionales el que mejor resultado demuestra, siendo sólo superado por las combinaciones de las redes neuronales convolucionales con redes de memoria de corto-largo plazo. [6] hace eco de la eficacia de las redes convolucionales y utiliza su implementación en tercera dimensión para analizar posturas a partir de videos, utilizando algoritmos que identifican articulaciones y las preservan a través del tiempo para hacer un seguimiento a través de los movimientos que hacen las personas al tomar estas posturas.

En el caso de la detección de poses de varios individuos, YogNet [7], de forma parecida a Anand et al. combina redes convolucionales con redes de memoria corto-largo plazo dividiendo el flujo de información en dos, enfocando la primera a la detección de características posturales y la segunda en las espacio-temporales.

## III. DISEÑO E IMPLEMENTACIÓN

En el diseño de este modelo se trabajó en dos fases, la primera se centró en la investigación y recolección del Set de datos, posteriormente se hizo un ligero preprocesado imágenes en el cual se intentó sacar las características más esenciales de las imágenes para que a partir de esto se diseñara modelo de Red neuronal con evolucionar el cual será capaz de distinguir entre cinco posturas de yoga diferentes. Éste modelo se sometió a un proceso de entrenamiento de aproximadamente 300 épocas.

### A. Conjunto de datos

Para el conjunto de datos se utilizó uno preexistente en la plataforma Kaggle conformada por más de 1500 imágenes distribuidas en cinco posturas de yoga: perro boca abajo (downdog), árbol (tree), tabla (plank), guerrero 2 (warrior2) y diosa (goddess) [8].

## B. Modelo

La red neuronal se conforma de una capa convolucional de entrada, dos convolucionales y una densa en la capa oculta y otra capa densa en la capa de salida, con cinco salidas correspondientes a las cinco categorías o posturas. Este se puede ver a más detalle en la Tabla 1.

Tipo de capa	Salida	# Parámetros
conv2D (capa de entrada)	(None, 146, 96, 32)	832
MaxPooling2D	(None, 73, 48, 32)	0
Conv2D	(None, 69, 44, 64)	51264
MaxPooling2D	(None, 34, 22, 64)	0
Conv2D	(None, 30, 18, 128)	204928
MaxPooling2D	(None, 15, 9, 128)	0
Flatten	(None, 17280)	0
Dense	(None, 1000)	17281000
Dense (capa de salida)	(None, 5)	5005
Parámetros totales: 17,543,029		
Parámetros entrenables: 17,543,029		
Parámetros no entrenables: 0		

TABLE I  
ARQUITECTURA DE LA RED NEURONAL.

## IV. RESULTADOS

A través del entrenamiento del modelo este demostró una eficacia del 99.65% sin embargo al contrastar con el conjunto de validación vemos que consigue acertar en un 94.48%. Como podemos ver en la Fig. 1, las curvas de entrenamiento y validación empiezan a divergir a partir de 160 épocas.

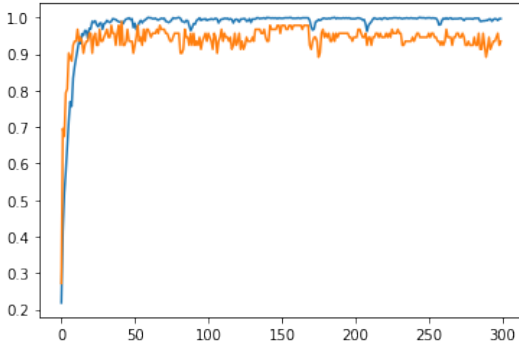


Fig. 1. Precisión del modelo, conjunto de entrenamiento (azul) vs conjunto de prueba (naranja).

Por otro lado, al evaluar el error, la función de pérdida nos arroja un valor de 0.0091 que contrasta fuertemente con el 0.4804 que arroja el conjunto de entrenamiento, valor que en ningún momento converge durante el proceso de entrenamiento y validación tal y como se puede ver en la fig. 2.

## V. CONCLUSIÓN Y TRABAJO FUTURO

Como ya se mencionó, después de las 160 épocas el modelo parece caer en un sobreentrenamiento, aún así, con una precisión del 94.48% el modelo cumple su objetivo al clasificar entre las posturas, sin embargo es necesario seguir

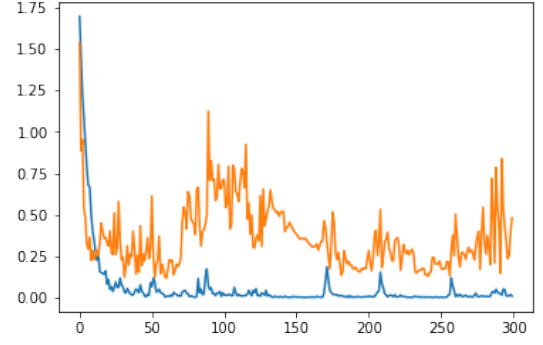


Fig. 2. Función de pérdida, conjunto de entrenamiento (azul) vs conjunto de prueba (naranja).

trabajando más en el pre-procesado de las imágenes pues con la puntuación de error demuestra que no diferencia con total certeza entre las categorías. Una detección de la posición de cada extremidad, tal y como otros autores han propuesto, podría ayudar a reducir este valor.

Por otra parte, el set de datos con el que se trabajó contaba con muy pocas muestras en general y pocas categorías. Con el fin de posteriormente desarrollar un sistema de aprendizaje de yoga es necesario un conjunto más robusto que incluya un mayor número de posturas y muestras para cada una de estas, tal y como podría ser el conjunto propuesto por Verma et al. [9]. Además, este sistema debe ser capaz de dar retroalimentación al usuario durante la ejecución de la actividad y no una vez se ha realizado dicha pose.

Por otra parte, no fue posible comprobar de primera mano la detección de posturas puesto que el grupo de investigación no es conocedor de yoga, por lo cual en un futuro trabajo se buscará a un practicante experimentado para poner a prueba el modelo.

## APPENDIX A

Todo el código relacionado a este trabajo se encuentra en [https://github.com/rakso-dev/TIA\\_proyecto](https://github.com/rakso-dev/TIA_proyecto).

## APPENDIX B

El acceso al conjunto de datos se encuentra en <https://www.kaggle.com/datasets/niharika41298/yoga-poses-dataset>.

## REFERENCES

- [1] Y. Kowsar, M. Moshtaghi, E. Velloso, L. Kulik, and C. Leckie, "Detecting unseen anomalies in weight training exercises," in *Proceedings of the 28th Australian Conference on Computer-Human Interaction, OzCHI '16*, (New York, NY, USA), p. 517–526, Association for Computing Machinery, 2016.
- [2] M. Sousa, J. a. Vieira, D. Medeiros, A. Arsenio, and J. Jorge, "Sleeveear: Augmented reality for rehabilitation using realtime feedback," in *Proceedings of the 21st International Conference on Intelligent User Interfaces, IUI '16*, (New York, NY, USA), p. 175–185, Association for Computing Machinery, 2016.
- [3] J. Palanimeera and K. Ponmozhi, "Classification of yoga pose using machine learning techniques," *Materials Today: Proceedings*, vol. 37, pp. 2930–2933, 2021.
- [4] V. Bhosale, P. Nandeshwar, A. Bale, and J. Sankhe, "Yoga pose detection and correction using posenet and knn," 2022.

- [5] V. Anand Thoutam, A. Srivastava, T. Badal, V. Kumar Mishra, G. Sinha, A. Sakalle, H. Bhardwaj, and M. Raj, "Yoga pose estimation and feedback generation using deep learning," *Computational Intelligence and Neuroscience*, vol. 2022, 2022.
- [6] T. K. K. Maddala, P. Kishore, K. K. Eepuri, and A. K. Dande, "Yoganet: 3-d yoga asana recognition using joint angular displacement maps with convnets," *IEEE Transactions on Multimedia*, vol. 21, no. 10, pp. 2492–2503, 2019.
- [7] S. K. Yadav, A. Agarwal, A. Kumar, K. Tiwari, H. M. Pandey, and S. A. Akbar, "Yognet: A two-stream network for realtime multiperson yoga action recognition and posture correction," *Knowledge-Based Systems*, p. 109097, 2022.
- [8] N. Pandit, "Yoga poses dataset," Oct 2020.
- [9] M. Verma, S. Kumawat, Y. Nakashima, and S. Raman, "Yoga-82: A new dataset for fine-grained classification of human poses," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 4472–4479, 2020.