Clasificación de Actividades Humanas con Redes Neuronales Convolucionales

Karla Cureño Vega

Maestría en Ciencia de Datos, Facultad de Ciencias Físico Matemáticas, Universidad Autónoma de Nuevo León

karla.curenov@uanl.edu.mx

Resumen: Se realizó la clasificación de actividades humanas con diferentes modelos utilizando redes neuronales convolucionales y *Transfer Learning* sobre un conjunto de 12,600 imágenes. El mejor modelo obtenido fue con una red *EfficientNet* con *fine-tuning* en las últimas 15 capas con un entrenamiento durante 53 *epochs* alcanzando una exactitud del 71 % sobre el set de prueba. © 2022

1. Introducción y Antecedentes

1.1. Problemática

La comprensión automática del comportamiento humano y su interacción con su entorno han sido un área de investigación activa en los últimos años, debido a su potencial para el desarrollo de la tecnología en diferentes dominios y aplicaciones.

Aplicaciones como la vigilancia, la recuperación de vídeo y la interacción persona-ordenador requieren métodos para reconocer las acciones humanas en diversos escenarios.

Por ejemplo, si una computadora fuera capaz de reconocer acciones con motivación sospechosa dentro de un sistema de videovigilancia podría alertar esto para poder evitar que sucedan situaciones de robo, secuestro, entre otras.

De igual manera, esta capacidad de clasificación de acciones humanas podría utilizarse para observar cámaras de tráfico enfocadas en peatones que están cruzando la calle y que tienen potencial de sufrir algún accidente.

Esta misma tecnología podría ser de mucha utilidad dentro de un entorno operativo en las fábricas, ya que mediante el análisis de vídeo de las cámaras operando dentro de las mismas podría generarse una alerta cuando haya un mal uso de las técnicas apropiadas para manejar maquinaria o cualquier tipo de movimiento que pueda generar un accidente laboral.

Por último, otra aplicación interesante es en el entrenamiento de robots que están diseñados para imitar acciones humanas ya que se puede utilizar una clasificación de acciones para comparar el movimiento que realiza el robot y el de un humano y poder concluir si el robot realiza de manera correcta la acción.

Como se ha establecido con los ejemplos anteriores, la solución de esta problemática tendría un gran impacto ya que tiene un potencial muy alto de aplicaciones.

1.2. Antecedentes

En el tema de clasificación de imágenes de acciones humanas ya existen antecedentes de modelos que han sido desarrollados de manera exitosa.

En Anitha et al. (2020) [1] los autores extraen las características de las imágenes y eso es lo que modelan, no en sí las imágenes, sino las características, por lo que ellos utilizan un clasificador KNN como modelo de clasificación. Realizan todo su código en `Matlab` y obtienen buenos resultados.

En Thurau y Hlavac (2008) [5] se clasifican las imágenes basándose únicamente en las poses que hay en las mismas, para esto utilizan un acercamiento muy matemático en el cual usan un algoritmo Histogram of Oriented Gradients para calcular las características de las poses y poder clasificar.

Finalmente, en Yu et al. (2020) [8] construyen diferentes redes convolucionales utilizando modelos preentrenados tales como VGG16 y ResNet50 para clasificar imágenes de acciones humanas. Subsecuentemente utilizaron el modelo DELWO para optimizar los pesos de los modelos construidos anteriormente y diseñaron un conjunto de aprendizaje profundo basado en el algoritmo de votación DELVS, todo esto para conseguir los mejores resultados de precisión en la clasificación de las imángenes.

2. Materiales y metodología

2.1. Conjunto de Datos

El conjunto de datos a utilizar consiste en 12,600 imágenes de actividades humanas. Las imágenes se encuentran en formato RGB y han sido etiquetadas correspondientemente a 15 diferentes clases. El set de datos se encuentra balanceado, por lo que en cada clase se tienen 840 imágenes. Este conjunto de datos fue recuperado de Kaggle [6] y las clases que se contemplan en el son:

- *calling*
- *clapping*
- *cycling*
- *dancing*
- *drinking*
- *eating*
- *fighting*
- *hugging*

- *laughing*
- *listening to music*
- *running*
- *sitting*
- *sleeping*
- *texting*
- *using laptop*

Como se puede notar, cada clase corresponde a una actividad humana diferente, por lo que en algunas imágenes se espera que haya más de un humano, mientras que en otras solo habrá una persona, y en otras puede existir un humano y un objeto en interacción. Se muestra en la figura 1 un ejemplo de imagen para cada clase o actividad humana.

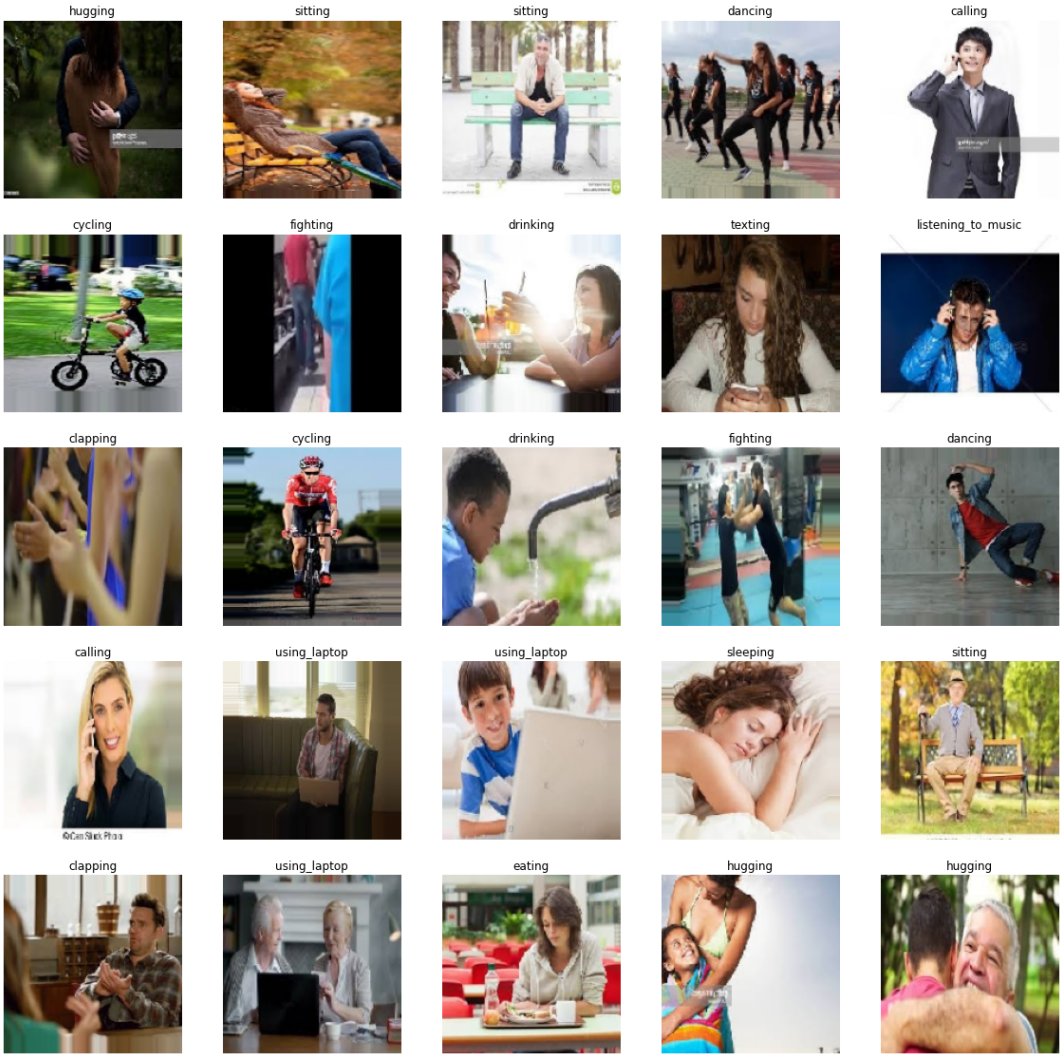

Figura 1: Imagénes ejemplo para cada clase del conjunto de datos.

Es importante mencionar que el conjunto de datos será dividido de manera aleatoria en 80 % para entrenamiento, 10 % para validación del modelo un último 10 % que se utilizará como set de prueba. El set de prueba no se considera para el entrenamiento del modelo, es un subconjunto separado desde el inicio que solo se utilizará para la evaluación del modelo.

2.2. *Metodología*

La carga de las imágenes se realizó con `ImageDataGenerator` ya que esta función permite realizar la carga de imágenes de manera óptima para los modelos de aprendizaje profundo.

La metodología para realizar el clasificador de acciones humanas se hizo con la utilización de las librerías `tensorflow`, `keras` de `Python` tanto como para desarrollar redes convolucionales desde cero, y para cargar modelos preentrenados con la biblioteca de imágenes de *Imagenet*.

Se utiizaron los modelos de redes neuronales convolucionales preentrenados *VGG16*, *ResNet50* y *EfficientNet* ya que al utilizar el algoritmo de *Transfer Learning* con los pesos de *ImageNet* se pueden obtener mejores resultados en clasificación de imágenes parecidas a las que ya existen dentro de de este diccionario de imágenes.

La red *VGG16* consiste en una red neuronal convolvucional que tiene 16 capas de profundidad y cuya arquitectura puede observarse en la figura 2.

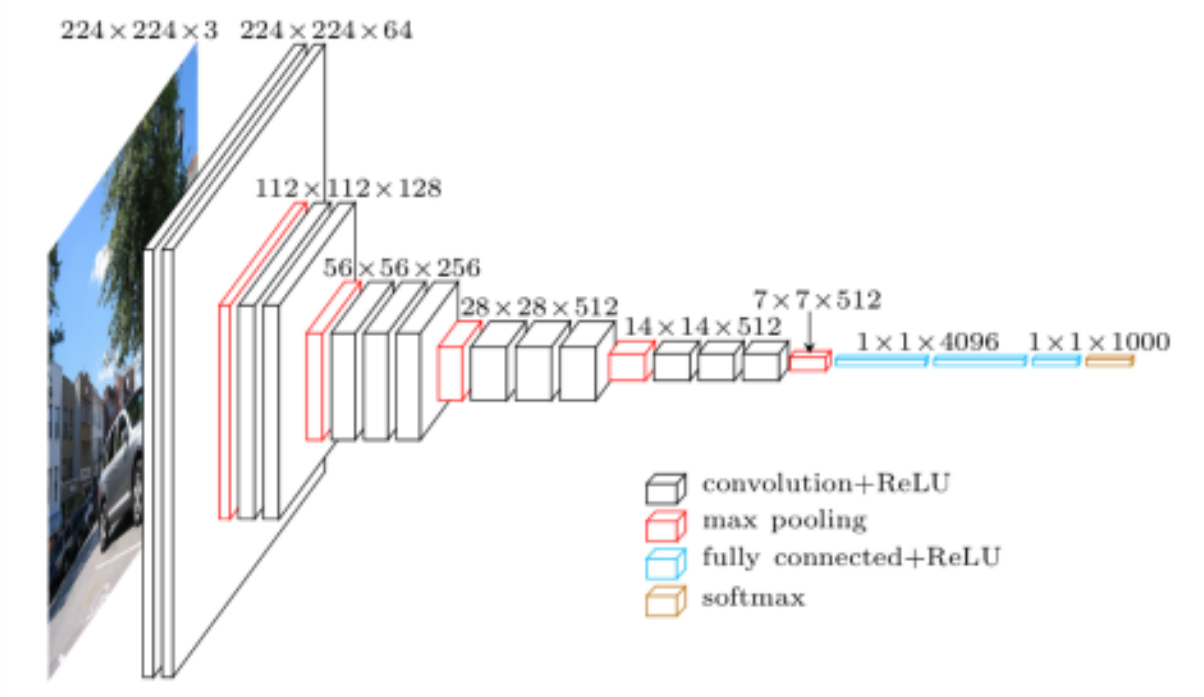

Figura 2: Arquitectura de la red VGG16 tomada de *TowardsDataScience*.

La red *ResNet50* es una red neuronal convolucional de 50 capas de profundidad. ResNet, abreviatura de Residual Networks, es una red neuronal clásica que se utiliza como columna vertebral en muchas tareas de visión computacional. El avance fundamental de ResNet es que

permitió entrenar redes neuronales extremadamente profundas. Se trata de una red neuronal innovadora que fue presentada por primera vez en He et al. (2015) [4]. Se puede observar un diagrama representando su arquitectura en la figura 4.

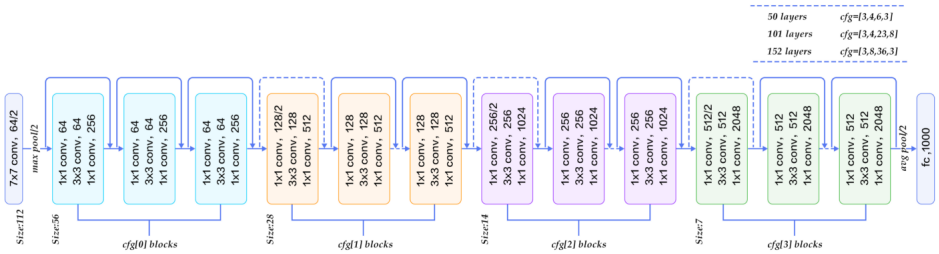

Figura 3: Arquitectura de la red ResNet50 tomada de *TowardsDataScience*.

La red *EfficientNet* es una arquitectura de red neuronal convolucional y un método de escalado que escala uniformemente todas las dimensiones de profundidad/anchura/resolución utilizando un coeficiente compuesto. A diferencia de la práctica convencional que escala arbitrariamente estos factores, el método de escalado de EfficientNet escala uniformemente la anchura, la profundidad y la resolución de la red con un conjunto de coeficientes de escalado fijos. Se puede observar un diagrama representando su arquitectura en la figura **??**

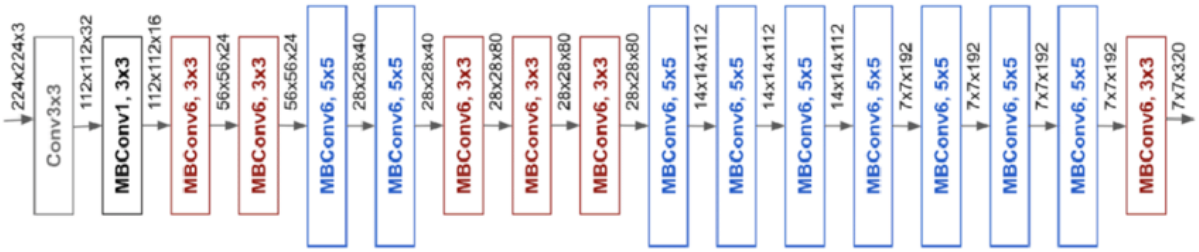

Figura 4: Arquitectura de la red EfficientNet tomada de *ResearchGate*.

Se compararon los diferentes modelos para elegir aquel que logre clasificar las imágenes con mayor precisión.

En la figura 5 se puede observar un diagrama que resume la metología y sus fases del marco de trabajo.

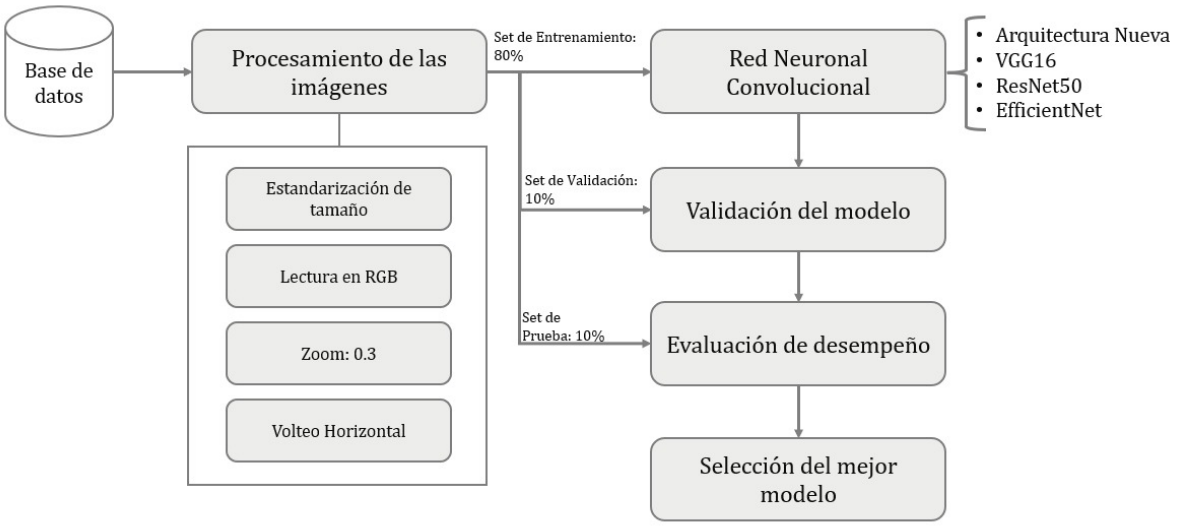

Figura 5: Diagrama de la metodología de trabajo.

3. Resultados

La experimentación con arquitecturas nuevas de redes convolucionales no preentrenadas resultó en una exactitud baja, alrededor del 10 % sobre el set de prueba, por lo que estos modelos se descartaron y se hablará sobre los utilizados con el algoritmo de *Transfer Learning* únicamente. Estos corresponden a las arquitecturas de las redes *VGG16*, *ResNet50* y *EfficientNet*.

En la figura 6 se pueden observar los resultados obtenidos durante el entrenamiento para la arquitectura *VGG16*. La exactitud obtenida con este modelo sobre el set de prueba fue del **53 %** con un entrenamiento durante 60 *epochs*.

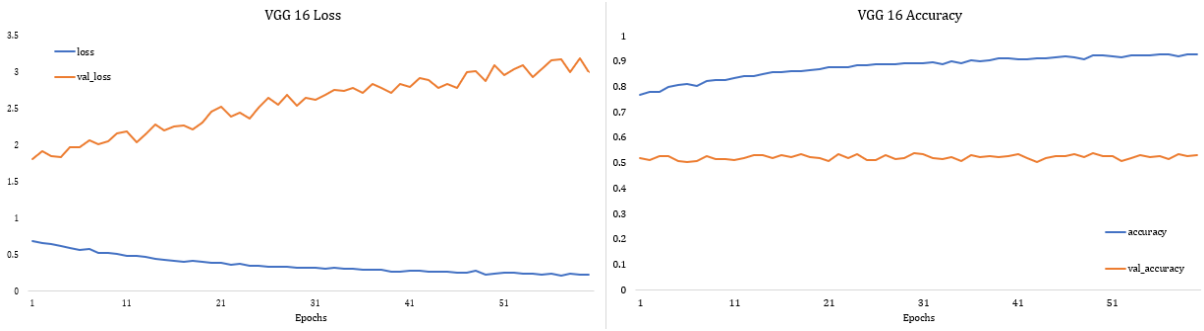

Figura 6: *Losss* y *Accuracy* durante el entrenamiento del modelo *VGG16*.

En la figura 7 se pueden observar los resultados obtenidos durante el entrenamiento para la arquitectura *ResNet50*. La exactitud obtenida con este modelo sobre el set de prueba fue del **61 %** con un entranamiento durante 33 *epochs* con todas las capas preentrenadas y 19 *epochs* con las últimas 15 capas sin preentrenar (*fine-tuning*), esto resulta en un entrenamiento durante 52 *epochs* en total.

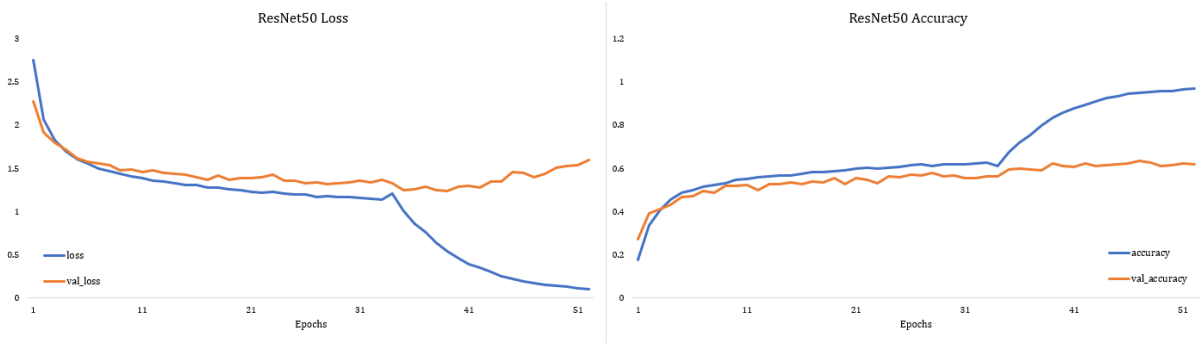

Figura 7: *Loss* y *Accuracy* durante el entrenamiento del modelo *ResNet50*.

En la figura 8 se pueden observar los resultados obtenidos durante el entrenamiento para la arquitectura *EfficientNet*. La exactitud obtenida con este modelo sobre el set de prueba fue del **72 %** con un entranamiento durante 31 *epochs* con todas las capas preentrenadas y 22 *epochs* con las últimas 15 capas sin preentrenar (*fine-tuning*), esto resulta en un entrenamiento durante 53 *epochs* en total.

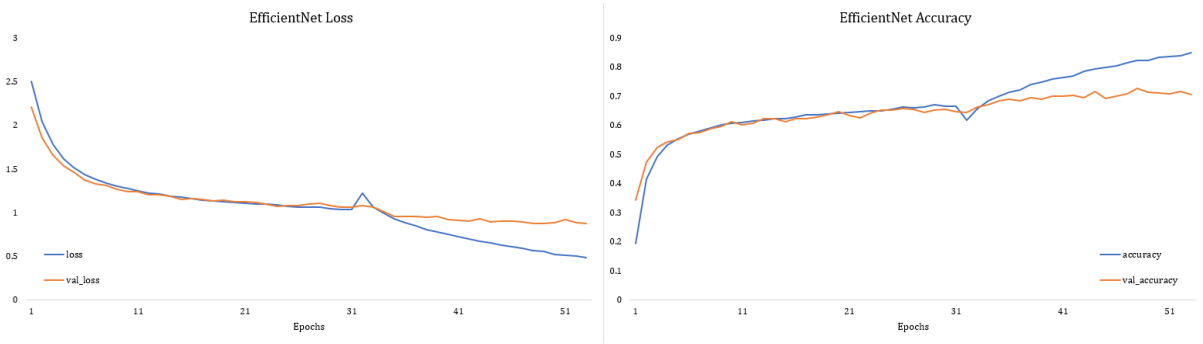

Figura 8: *Loss* y *Accuracy* durante el entrenamiento del modelo *EfficientNet*.

Debido a que el modelo desarrollado con la arquitectura de *EfficientNet* fue el que presentó mejor exactitud, se selecciona como el mejor modelo de la experimentación.

En la tabla 1 se observa el reporte de clasificación del mejor modelo y en la figura 9 su matriz de confusión.

	precision	recall	f1-score	support
calling	65 %	49 %	56 %	84
clapping	79 %	60 %	68 %	84
cycling	93 %	94 %	93 %	84
dancing	73 %	75 %	74 %	84
drinking	63 %	70 %	66 %	84
eating	92 %	87 %	90 %	84
fighting	79 %	77 %	78 %	84
hugging	60 %	68 %	64 %	84
laughing	72 %	73 %	72 %	84
listening to music	55 %	63 %	59 %	84
running	92 %	85 %	88 %	84
sitting	54 %	62 %	57 %	84
sleeping	81 %	77 %	79 %	84
texting	57 %	51 %	54 %	84
using laptop	58 %	70 %	63 %	84
accuracy	71 %	71 %	71 %	**71 %**
macro avg	72 %	71 %	71 %	1260
weighted avg	72 %	71 %	71 %	1260

Tabla 1: Reporte de Clasificación sobre el set de prueba con el modelo *EfficientNet*.

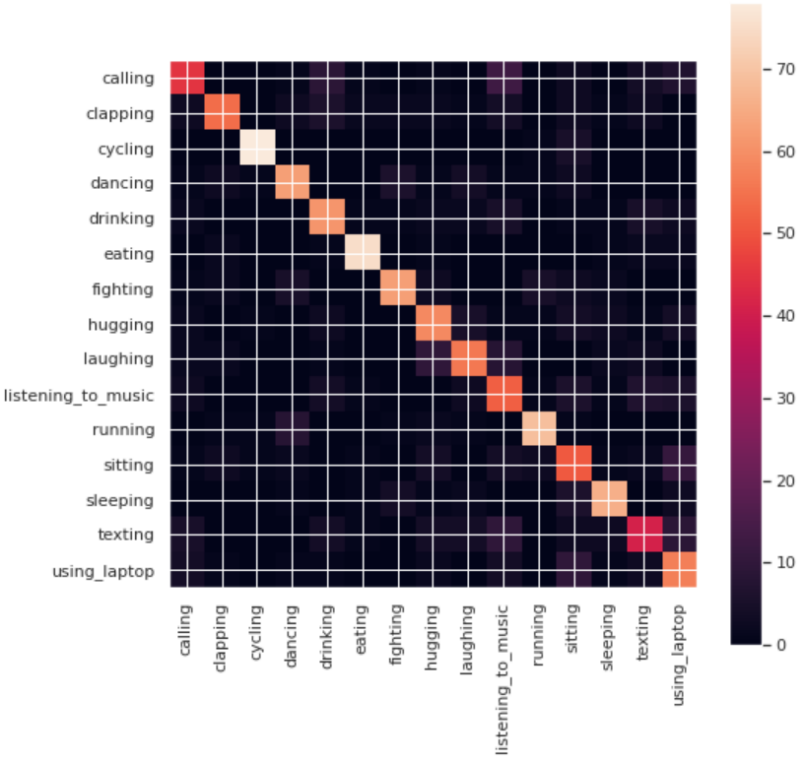

Figura 9: Matriz de Confusión sobre el set de prueba para el modelo *EfficientNet*.

4. Discusión de Resultados

Los resultados mostrados en la sección anterior demuestran que el modelo obtenido con la red *EfficientNet* es el modelo analizado con mayor exactitud para el conjunto de datos de estudio.

Este modelo alcanza una exactitud del 71 % sobre el conjunto de datos de prueba y como se observa en el reporte de clasificación 1 en algunas clases tiene una muy buena precisión llegando hasta el 93 % en la actividad de *cycling*.

Sin embargo, se puede notar también que hay algunas actividades que son más difíciles de clasificar para el modelo, entre ellas se encuentran *sitting*, *listening to music*, *texting* y *using laptop*. Al observar las imágenes de estas actividades en la figura 1 se puede ver que son justamente las actividades donde hay una persona interactuando con un objeto, por lo que puede ser que el hecho de que haya algo más que el humano dentro de la imagen haga que el modelo pueda confundirse o equivocarse en la clasificación.

Por otro lado, una característica positiva de este modelo es que observando la gráfica de *accuracy* que se muestra en 8 se puede ver que la exactitud alcanzada sobre el conjunto de validación es del 71 % y que al evaluar el modelo sobre el set de prueba esta métrica llega al mismo resultado, lo que demuestra que el modelo está entrenado de manera correcta y no hay sobreajuste.

Comparando resultados con aquellas metodologías similiares a la propuesta y mencionadas en la sección de Antecedentes, tal como lo es mostrado en la tabla 2 se puede decir que el resultado del modelo propuesto es muy bueno ya que el nivel de exactitud que tiene es muy parecido e incluso un poco mejor que el de las metodologías referenciadas.

Método	Número de Imágenes en Conjunto de Datos	Mejor Exactitud
Thuru y Hlavac (2008)	130	70 %
Yu et al. (2020)	968	68 %
Propuesto	**12,600**	**71 %**

Tabla 2: Comparación de resultados con antecedentes con métodos similares al propuesto en este trabajo.

5. Conclusiones y trabajo a futuro

Se logró realizar la clasificación de actividades humanas utilizando diferentes redes neuronales convolucionales, siendo la de mejor resultado la de *EfficientNet* con una exactitud del 71 %.

El resultado de exactitud es el mismo sobre el set de validación y el de prueba, lo que demuestra que el modelo es confiable y no tiene sobreajuste.

Este resultado de exactitud es bueno comparándolo con aquellos de los antecedentes referenciados en este trabajo.

Como trabajo a futuro, podría mejorarse este modelo agregando una mayor cantidad

de imágenes para el entrenamiento del mismo, esto puede hacerse capturando imágenes de diferentes actividades humanas de manera natural o buscando imágenes de este tipo en internet.

De igual manera, con una mayor capacidad de cómputo se podría utilizar una red neuronal más profunda con la finalidad de lograr una mejor clasificación sobre las imágenes.

Referencias

1. Anitha, U., Narmadha, R., Sumanth, D. R. & Kumar, D. N. (2020). Robust human action recognition system via image processing. Procedia Computer Science, 167, 870-877. doi:10.1016/j.procs.2020.03.426
2. Cureno, K. (2022). Clasificación de Actividades Humanas usando Redes Neuronales Convolucionales - Notebook. Recuperado de https://github.com/karlacuv/MCD_Procesamiento/blob/main/ProyectoFinal.ipynb
3. Guo, G & Lai, A. (2014). A survey on still image based Human Action Recognition. Pattern Recognition, 47(10), 3343-3361. doi:10.1016/j.patcog.2014.04.018
4. He, K., Zhang, X., Ren, S., amp; Sun, J. (2016). Deep residual learning for image recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). doi:10.1109/cvpr.2016.90
5. Maity, S., Bhattacharjee, D. & Chakrabarti, A. (2016). A novel approach for human action recognition from Silhouette Images. IETE Journal of Research, 63(2), 160-171. doi:10.1080/03772063.2016.1242383
6. Nagadia, M. (2022). Human Action Recognition (HAR) Dataset. Recuperado en Julio, 2022, de https://www.kaggle.com/datasets/meetnagadia/human-action-recognition-har-dataset
7. Thurau, C. & Hlavac, V. (2008). Pose primitive based human action recognition in videos or still images. 2008 IEEE Conference on Computer Vision and Pattern Recognition. doi:10.1109/cvpr.2008.4587721
8. Yu, X., Zhang, Z., Wu, L., Pang, W., Chen, H., Yu, Z. & Li, B. (2020). Deep Ensemble Learning for Human Action Recognition in still images. Complexity, 2020, 1-23. doi:10.1155/2020/9428612