ISSN 1870-4069

Reconocimiento de acciones de empaquetado usando redes CNN-biLSTM y optimización bayesiana

Alberto Angulo Landeros[1], Luis A. Castro[1], Jessica Beltrán-Márquez[2]

[1] Instituto Tecnológico de Sonora, Ciudad Obregón, México

[2] Universidad Autónoma de Coahuila, Centro de Investigación en Matemáticas Aplicadas, México

luis.castro@acm.org, alberto.angulo242400@potros.itson.edu.mx, jessicabeltran@uadec.edu.mx

Resumen. En la actualidad, el empaquetado de productos aún depende principalmente de trabajadores manuales. Para garantizar una respuesta rápida a las demandas cambiantes de los clientes, se espera que la tendencia continúe. Por lo tanto, cuantificar el trabajo realizado es de suma importancia para la optimización de procesos. La heterogeneidad tanto del tamaño y forma de los productos a empacar, como la variabilidad de cómo los empleados empacan artículos dificultan el proceso de reconocimiento de la actividad en esta área. Para resolver este problema de reconocimiento de actividad humana (Human Activity Recognition) se han utilizado varios enfoques. Recientemente, se han utilizado métodos de aprendizaje profundo como RNN y LSTM para esta tarea. Sin embargo, estas arquitecturas no logran obtener buenos resultados cuando se trata de capturar dependencias a largo plazo en datos de series temporales, más aún cuando las actividades son secuenciales. En este trabajo, se propone una arquitectura de red convolucional con memoria a largo y corto plazo bidireccional para reconocer acciones de empaquetado en un entorno industrial. Además, se implementó una optimización bayesiana para lograr encontrar la mejor configuración de hiperparámetros. La arquitectura se evaluó utilizando el conjunto de datos Openpack logrando 93.15% de Valor F1, superando los resultados de las arquitecturas de referencia.

Palabras clave: HAR, empaquetado de productos, optimización bayesiana, redes neuronales convolucionales, redes de memoria a largo y corto plazo.

Recognition of Packet Actions Using CNN-biLSTM Networks and Bayesian Optimizationet

Abstract. Currently, product packaging processes still depend mainly on manual workers. To ensure a quick response to changing customer demands, the trend is expected to continue. Therefore, quantifying the work done is paramount to

optimizing processes. The heterogeneity of the size and shape of the packed products and the variability of how employees pack items make it difficult to recognize the activity in this area. Various approaches have been used to solve this Human Activity Recognition (HAR) problem. Recently, deep learning methods such as RNN and LSTM have been used for this task. However, these architectures fail to perform well when it comes to capturing long-term dependencies on time series data, even more so when activities are sequential. This work proposes a convolutional network architecture with bidirectional long short-term memory to recognize packaging actions in an industrial environment. In addition, a Bayesian optimization was implemented to find the best hyperparameter configuration. The architecture was evaluated using the Openpack data set, achieving 93.15% F1-Value, surpassing the results by the reference architectures.

Keywords: HAR, product packaging, Bayesian optimization, convolutional neural networks, long short-term memory.

1. Introducción

El reconocimiento de actividades humanas (Human Activity Recognition, HAR) es un campo enfocado en identificar las acciones realizadas por una persona [1]. Debido al progreso en tecnologías de sensores y computación ubicua, el HAR ha expandido su aplicación en diversas áreas. Actualmente, el HAR se utiliza ampliamente en sectores como la atención médica [2], monitoreo de empleados [3], interfaces hombre-máquina [4, 5], entrenamiento deportivo [6], vigilancia [7], entre otros.

El avance acelerado en las tecnologías de sensores e informática ubicua ha impulsado la creciente popularidad del uso de datos de sensores en el reconocimiento de actividades humanas. En el entorno industrial, HAR tiene varias aplicaciones, entre estas se encuentra la identificación de la manera en que los empleados realizan las actividades, lo que puede servir para la mejora y optimización de procesos.

Por ejemplo, el personal que labora en centros logísticos en el área de empaquetado realiza una serie de actividades manuales secuenciales para empacar artículos. Algunos de los aspectos que pueden ser de interés es la identificación de posturas inadecuadas de los empleados del área de empaquetado, que puede llevar a lesiones y afectaciones consecuentes en la productividad.

Mediante HAR, es posible reconocer acciones y la forma en que se realizan, para identificar a los empleados que no realizan adecuadamente el empaquetado, señalar los errores que se cometen y generar recomendaciones para mejorar las posturas. Asimismo, mediante HAR se pueden identificar otras acciones relacionadas con la eficiencia de la producción, como el correcto seguimiento de los protocolos de empaquetado, las anomalías en el empaquetado, las técnicas de empaquetado más eficientes, entre otras.

Recientemente se han hecho investigaciones tratando de abordar este problema. Por ejemplo, en [8] se propuso un enfoque para reconocer cuatro actividades diferentes utilizando datos obtenidos de un acelerómetro triaxial y un giroscopio. Las actividades fueron martillar, atornillar con un destornillador, usar una llave inglesa y fijar tornillos con un taladro eléctrico. Se utilizó el método del vecino más cercano (kNN) para clasificar los datos.

Tabla 1. Detalles de conjunto de datos Openpack para reconocimiento de actividades humanas.

Elementos	Detalles
Tipo	Reconocimiento de trabajo de empaquetado
Participantes	16
Tasa de muestreo	IMU (Acc, Gyro, Ori): 30Hz
	Empática E4 (BVP, EDA): 64HZ y 4Hz
	Sensores Empática E4 (ACC): 32Hz
Actividades (clases)	10 principales + 32 secundarias
No. de datos obtenidos	20,129 operaciones de trabajo
	52,529 acciones
Modalidad	D+Keypoints+LiDAR+Acc+Gyro+Ori+EDA+BVP+Temp
Duración de la grabación	53h50m

D: Depth, Acc: acelerómetro, Gyro: giroscopio, Ori: sensor de orientación, EDA: actividad electrodérmica, BVP: pulso de volumen sanguíneo, Temp: temperatura. Keypoints: puntos cla del sensor Kinect.

En otro trabajo se implementó una arquitectura de redes convolucionales para identificar acciones realizadas en un contexto de logística [9]; Los autores evaluaron el modelo utilizando un conjunto de datos propio llamado LARa. Para ello, aplicaron una técnica de ventana deslizante junto con redes neuronales convolucionales combinada con capas lineales.

En la última capa de la arquitectura, se implementaron dos funciones de activación: softmax y sigmoid. Finalmente, se utilizó la función de pérdida de entropía cruzada, fusionando ambas salidas para optimizar el rendimiento del modelo obteniendo un Valor-F1 de 64.43%.

Algunos de los retos existentes en el área de HAR son el reconocimiento de actividades a partir de datos multimodales, el reconocimiento adecuado de actividades formadas por acciones secuenciales, y el reconocimiento de actividades en diferentes granularidades.

En el caso específico de problemas orientados a empaquetado de artículos, un reto importante se debe a la complejidad de las actividades realizadas, a la heterogeneidad del tamaño y forma de los artículos a empaquetar, así como la variabilidad de la manera en que los empleados empaquetan artículos.

Asimismo, otro reto se debe a la similitud en la manera en que se realizan actividades distintas. Por ejemplo, cuando las personas cierran una caja es similar a cuando se agrega una etiqueta a la caja. Este trabajo se centra en la utilización de técnicas de HAR orientadas a reconocer actividades de empaquetado en un entorno industrial utilizando el conjunto de datos Openpack [10].

Este conjunto de datos incluye datos multimodales como datos de sensores IMU (Acc, Gyro, Ori), puntos clave de sensor Kinect, visión de profundidad y provee una granularidad de categorías de actividades mayor en comparación con otros conjuntos de datos disponibles como InHARD [11] y LARa [9]. Para la clasificación de las

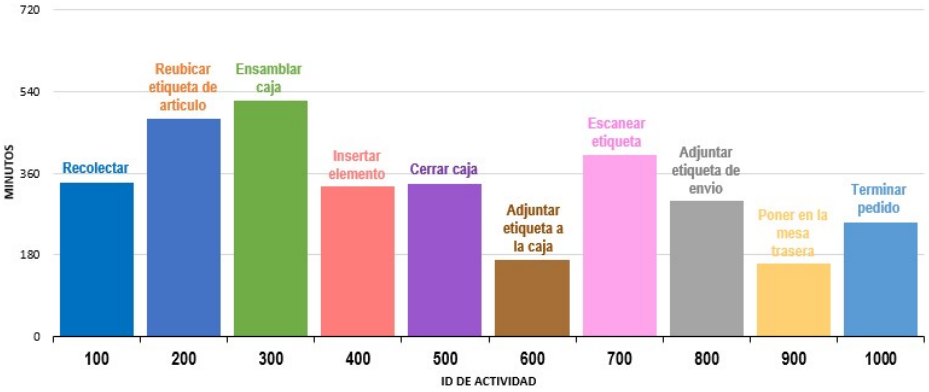

Fig. 1. Distribución de duración total de grabación de cada actividad. Imagen basada en [10].

actividades, se propone una variante de red convolucional profunda bidireccional CNN-biLSTM que se aplica al escenario de empaquetado de productos.

Además, se utiliza Optimización Bayesiana para encontrar la mejor arquitectura e hiperparámetros de la red propuesta. Finalmente, se evaluó la red propuesta utilizando diferentes combinaciones de sensores y conjuntos de usuarios.

2. Trabajo relacionado

El reconocimiento de la actividad humana es un problema basado en series temporales. La evaluación y el análisis de estas señales para el reconocimiento de actividades humanas es de especial interés para realizar optimizaciones en la industria donde el trabajo manual sigue siendo dominante.

Se han desarrollado métodos para clasificar los movimientos humanos. Un ejemplo de aporte en el área de HAR es un trabajo orientado a reconocer las actividades y movimientos humanos en la preparación de pedidos [12]. En [12], los autores utilizaron datos de tres unidades de medición inercial (IMU) que usaron los trabajadores mientras hacían actividades en dos escenarios de preparación de pedidos comparables (A y B).

Ambos escenarios se operaron de manera manual, las mercancías se almacenan en estantes, el trabajador se encargaba de recolectar los pedidos por el almacén y cada artículo se almacenó en una caja dedicada. En el escenario A, los pedidos se proporcionaron en papel y en B se utilizaron dispositivos portátiles con conexión a WiFi.

Se utilizaron características estadísticas en el dominio del tiempo en segmentos utilizando el enfoque de ventana deslizante. Se evaluaron tres clasificadores: una máquina de vectores de soporte, un clasificador Naive Bayes y un bosque aleatorio utilizando validación de 3 k-fold, los resultados fueron 67.3, 67.7 y 72.6 respectivamente, mostrando que el bosque aleatorio muestra el rendimiento más estable.

En otro trabajo, se propuso una arquitectura de red neuronal profunda utilizando datos secuenciales de múltiples unidades de medida inercial (IMU) [13]. Se evaluaron los datos de acelerómetro, giroscopio y magnetómetro. Los autores reportaron una

Tabla 2. Detalles de escenarios en conjunto de datos Openpack.

ID	Descripción
ES01	Los participantes siguieron las instrucciones lo más fielmente posible. La lista de artículos en un pedido se basó en hojas de pedidos reales, pero se limitó la variedad de artículos en un pedido a 54.
ES02	Los participantes tuvieron libertad para alterar el procedimiento de las operaciones según su criterio. También se redujeron las probabilidades de incluir artículos muy grandes o pequeños en un pedido en comparación con la ES01, y se agregaron 21 artículos nuevos.
ES03	Se introdujeron situaciones/acciones irregulares al ES02, como cajas de envío ya ensambladas que podían ser utilizadas por los trabajadores, la inclusión de artículos pequeños en bolsas de papel, y la posibilidad de que un sujeto llevara varios pedidos consecutivos de artículos pequeños de la mesa trasera al banco de trabajo al mismo tiempo.
ES04	Se implementó una alarma auditiva en el ES03 para simular un tiempo de trabajo ocupado y se establecieron alarmas periódicas (con un intervalo de 30-45 segundos) cuando el tiempo transcurrido de un periodo excedía el 80% de la duración promedio de un periodo de trabajo previamente registrado.

mejoría de hasta 2% de exactitud en la clasificación comparado con enfoques tradicionales (p.ej., bosque aleatorio), así como otras arquitecturas de redes neuronales.

Además, entre los métodos HAR que han sido propuestos, se encuentran los que son basados en redes convolucionales (Convolutional Neural Network, CNN) en donde la extracción de características es parte de la arquitectura de la red neuronal [14, 15]. Se propone un enfoque basado en CNN para la clasificación de actividades humanas utilizando datos provenientes de diferentes tipos de sensores colocados en el cuerpo de las personas.

El enfoque propuesto se probó en tres diferentes conjuntos de datos: Opportunity [16, 17], PAMAP2 [18] y Order Picking [19]. Los tres conjuntos de datos presentan un tipo y cantidad distinta de actividades y desbalance de clases. Los autores evaluaron la red propuesta utilizando series de tiempo multicanal adquiridas de sensores corporales IMU. Se logró un mejor resultado utilizando la red CNN-IMU propuesta comparado a una red CNN base.

Además, se investigó el efecto de utilizar operaciones de agrupación máxima, ya que esta operación podría no conservar la información como se sugiere en [20], para secuencias relativamente largas, las CNN que contienen operaciones de agrupación máxima muestran mejor resultados.

En [10] se propuso un conjunto de datos multimodal sobre el reconocimiento de actividades laborales en un entorno industrial. Además, se propuso un nuevo modelo de reconocimiento LTS-Net para la clasificación de series temporales que utiliza lecturas provenientes de dispositivos de internet de las cosas (IoT). Para las CNN es difícil extraer la dependencia a largo plazo dentro de una serie temporal, lo que dificulta mejorar el rendimiento del modelo.

Una propuesta de solución son las llamadas redes de memoria a largo y corto plazo (LSTM) [21], las cuales se han empleado en HAR debido a sus ventajas para extraer dependencias a largo plazo dentro de series temporales. En [22] se propuso una red

Tabla 3. Detalles de participantes y sesiones en conjunto de datos OpenPack.

Participantes					Sesiones				
ID	Sexo	Edad	Mano Dominante	Experiencia	S0100	S0200	S0300	S0400	S0500
U0101	F	-	D	-	ES01	ES01	ES01	ES01	ES01
U0102	F	-	D	-	ES01	ES01	ES01	ES01	ES01
U0103	F	50	D	6 meses	ES01	ES01	ES01	ES01	ES01
U0105	F	30	D	4 años	ES01	ES01	ES01	ES01	ES01
U0106	F	40	D	1 mes	ES01	ES01	ES01	ES01	ES01
U0107	F	40	D	3 años	ES01	ES01	ES01	ES01	ES01
U0109	M	30	I	6 meses	ES01	ES01	ES01	ES01	ES01
U0110	F	40	D	10 meses	ES01	ES01	ES01	ES01	ES01
U0111	F	50	D	2 años	ES01	ES01	ES01	ES01	ES01
U0205	F	30	D	4 años	ES02	ES02	ES03	ES03	ES04
U0202	F	40	D	3 años	ES02	ES02	ES03	ES03	ES04
U0210	F	50	D	3 meses	ES02	ES02	ES03	ES03	ES04

neuronal profunda que combina capas convolucionales con memoria a corto plazo para el reconocimiento de actividad humana, el enfoque propuesto es capaz de aprender la dinámica temporal en varias escalas de tiempo aumentando la exactitud obtenida.

El uso de capas convolucionales con memoria a corto y largo plazo ha ayudado a aumentar la precisión en diferentes conjuntos de datos HAR. Sin embargo, las investigaciones actuales se han enfocado en reconocer actividades de uso cotidiano como caminar, subir escaleras, bajar escaleras, o andar en bicicleta. Existe muy poca literatura relacionada a actividades en el área del empaquetado de productos en el área industrial [9, 10, 12, 13, 19].

3. Métodos

En este trabajo, se propone una red CNN-biLSTM que nos permite la extracción de la dependencia de tiempo tanto hacia atrás como hacia adelante y al tratarse de actividades secuenciales nos permite predecir la actividad de interés no solo de la actividad anterior, sino también de la siguiente actividad.

3.1. Conjunto de datos

Se seleccionó el conjunto de datos Openpack [10] para este estudio debido a que es considerado el conjunto de datos más completo en el reconocimiento de actividad humana en la industria, específicamente en el área del empaquetado de productos. Este conjunto de datos contiene una gran cantidad de datos de sensores y se compone de registros de 16 participantes realizando actividades de empaquetado en entornos industriales.

La tabla 1 proporciona más información detallada sobre los elementos que forman parte y describen el conjunto de datos Openpack. La construcción del conjunto de datos Openpack siguió un documento de instrucciones utilizado en un centro de logística real, el cual especifica la secuencia de acciones que debe realizar el trabajador durante el empaquetado de productos.

Tabla 4. Configuraciones utilizadas para evaluar arquitectura CNN-biLSTM para Entrenamiento (E), Validacion (V) y Pruebas (P).

	Subconjunto 1		**Subconjunto 2**	
	Participante	Sesión	Participante	Sesión
E	U0102	S0100, S0200, S0300	U0101, U0102, U0103, U0105, U0106, U0107, U0109, U0110	S0100, S0200, S0400, S0500
V	U0102	S0400	U0101, U0103, U0105, U0107, U0109, U0111, U0205	S0300
P	U0102	S0500	U0102, U0106, U0202, U0210	S0300

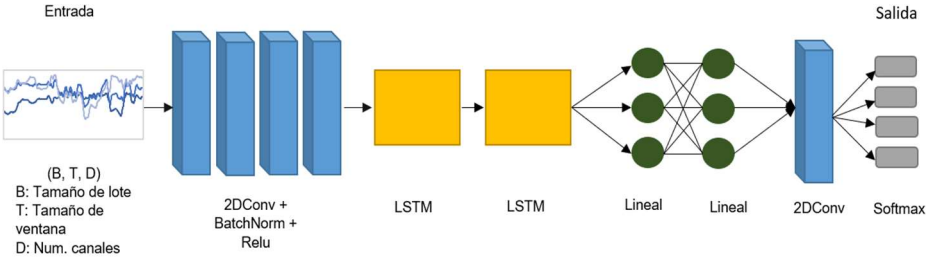

Fig. 2. Arquitectura del modelo para CNN-biLSTM.

Los creadores de Openpack utilizaron estas acciones para etiquetar el conjunto de datos. En la Fig. 1 se muestran las clases de operaciones de empaquetado, junto con la cantidad de minutos capturados correspondientes a cada una de ellas.

Como se observa en la Fig. 1, las clases están desbalanceadas. La actividad de reubicar la etiqueta del artículo y ensamblar caja son las actividades con mayor número de muestras. Las muestras son calculadas multiplicando los segundos de grabación y la tasa de muestreo de cada sensor.

Los participantes realizaron las acciones siguiendo 4 escenarios preestablecidos. En la Tabla 2 se muestran los detalles de escenarios utilizados para la obtención del conjunto de datos. En la Tabla 3 se muestra información sobre los participantes, las sesiones y los escenarios que realizaron.

La propuesta en este estudio se enfoca en utilizar los datos de los sensores Atr (acc, gyro, quaternion) y E4 (acc, BVP, EDA, temperatura) con los cuales se realizaron diferentes experimentos. Los sensores Atr se ubicaron en ambas muñecas y ambos brazos, los sensores E4 se ubicaron en ambas muñecas.

3.2. Preparación de datos

Se aplicaron diferentes técnicas de preprocesamiento sobre los datos. Para reducir el ruido, se aplicó el filtro Kalman utilizando la librería Pykalman[1] aplicando una covarianza de observación de 0.1 y covarianza de transición de 0.01.

[1] Pykalman, pykalman.github.io/, último acceso 17/04/2023

Tabla 5. Lista de hiperparámetros seleccionados.

Escenario	Hiperparámetros	Valores seleccionados
Procesamiento de datos	Tamaño de ventana	1800
Entrenamiento	Optimizador	Adam
	Tamaño de lote (Batch Size)	12
	Tasa de Aprendizaje (Learning Rate)	0.0006
	Caída de peso (Weight Decay)	0.0005
	Épocas	50

Tabla 6. Combinaciones de sensores utilizados para evaluación.

Combinación	C1	C2	C3	C4	C5	C6
Sensor	Acc, E4Acc	Gyro	Acc, E4Acc, Gyro	Acc, E4Acc, Gyro, Ori	Acc, E4Acc, Gyro, Ori, Bvp	Acc, E4Acc, Gyro, Ori, Bvp, Eda

Debido a que los datos fueron capturados en condiciones realistas y los sensores que se usan en los sujetos son inalámbricos, algunos datos fueron perdidos durante el proceso de recopilación y la sincronización. Para evitar esto, los datos se preprocesaron mediante la técnica de interpolación.

Otro problema que se detectó en los datos es un desfase en los tiempos de los datos obtenidos. Uno de los motivos se debe a que los sensores capturan datos usando diferentes frecuencias de muestreo. Otro motivo es que al momento de la captura algunos sensores empezaron a grabar antes y terminaron antes que los otros sensores en algunas sesiones.

Por ejemplo, en la sesión S0100 del usuario U0102, los datos del sensor Atr están adelantados un segundo, y en la sesión S0100, el sensor E4 está adelantado un segundo. Para abordar este problema, se realizó un preprocesamiento que consistió en remuestrear los datos a 25Hz y sincronizar los datos utilizando las épocas de Unix.

3.3. Separación de datos

Debido a la complejidad de los datos, se trabajó con un subconjunto de datos (denominado “Subconjunto 1”) para encontrar los mejores hiperpárametros, ya que el costo computacional de entrenamiento es elevado y además requiere una cantidad considerable de tiempo.

Posteriormente, se utilizó un conjunto de datos más amplio (denominado “Subconjunto 2”) para entrenar y evaluar la mejor configuración de hiperparámetros. En la Tabla 4 se detallan los participantes y sesiones que formaron parte de cada experimento. Es importante mencionar que el conjunto de datos Openpack incluye datos sin etiquetar o con numerosos valores faltantes, los cuales fueron excluidos.

3.4. Arquitectura propuesta

La estructura de la red propuesta en este trabajo (CNN-biLSTM) es una variación de la arquitectura descrita en [23], y combina capas convolucionales, lineales y

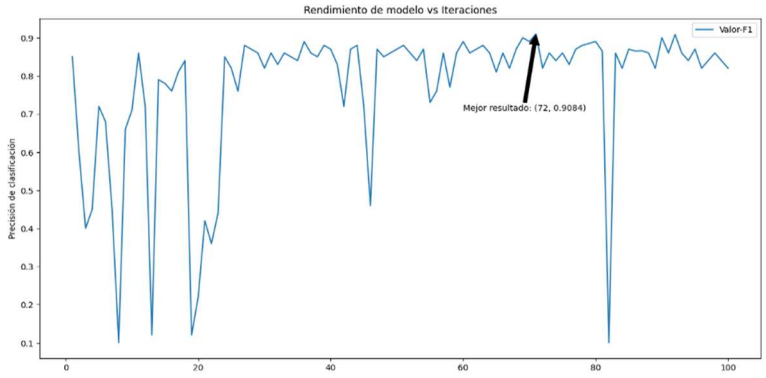

Fig. 3. Rendimiento del modelo vs iteraciones para obtener hiperparámetros de modelo propuesto utilizando el Valor-F1 con datos de configuración de Subconjunto 1 y sensor Gyro.

recurrentes. Las capas convolucionales se encargan de extraer características espaciales y proporcionar representaciones abstractas de los datos de entrada en mapas de características, mientras que las capas recurrentes aprenden las dependencias a largo plazo tanto hacia adelante como hacia atrás. La arquitectura propuesta consta de nueve capas (Fig. 2).

Como se observa en la Fig. 2, los datos preprocesados ingresan a la red CNN-BiLSTM propuesta, en donde el primer paso consiste en 4 capas que cuentan con 99 filtros encargados de extraer características espaciales. Entre cada capa, se encuentra una normalización por lotes estándar que actúa como regularizador, así como una función de activación ReLU.

Le siguen dos capas LSTM bidireccionales con 234 neuronas, que se encargan de obtener características temporales. Para evitar el sobreajuste se utilizan capas de abandono con 0.33 y 0.16 respectivamente. La segunda capa LSTM usa como entrada la salida de la capa anterior. Luego se agregan dos capas lineales totalmente conectadas con 330 y 223 neuronas, respectivamente.

Por último, la salida del modelo está dada por una capa de salida (una capa Conv con una función de activación softmax). Las variaciones implementadas a la arquitectura propuesta en comparación con la de referencia incluyen el uso de LSTM bidireccionales, lo cual permite predecir la actividad de interés tanto a partir de la actividad anterior cómo a partir de la actividad siguiente.

Además, se agregaron dos capas lineales, las cuales mejoran la capacidad del modelo para aprender representaciones más complejas de los datos. De esta manera, se logra mejorar la capacidad de generalización del modelo y evitar el sobreajuste.

3.5. Entrenamiento

La arquitectura propuesta se implementó en Pytorch y se utilizó la función de pérdida entropía cruzada para la optimización de la red, tomando en cuenta las 10 clases de actividad descritas en la Fig. 1.

La entrada de la red consiste en una secuencia de datos formada por series de tiempo extraídas desde los datos de los sensores Atr (Acc, Gyro, Ori) y E4 (Acc, BVP, EDA)

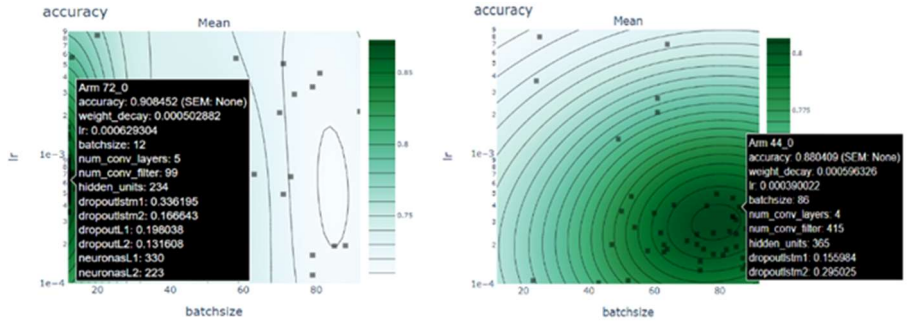

Fig. 4. Mejores hiper parámetros obtenidos utilizando optimización bayesiana para CNN-biLSTM (izquierda); Mejores hiperparámetros obtenidos utilizando optimización bayesiana para DeepConvLSTM (derecha).

utilizando un enfoque de ventana deslizante compuesto por varios canales de sensores. Para demostrar la eficiencia del modelo, tanto en el entrenamiento como en la prueba, los datos se segmentan en tamaños de lotes de 12 datos por segmento.

Los datos recibidos por la primera capa convolucional son de la forma (B, CH, T, 1) donde B es el tamaño del lote, CH es el número de canales de entrada, T es el tamaño de la ventana que en este caso fue de 30 muestras por segundo, dando un tamaño de 1800.

Después de ser procesados por las cuatro capas convolucionales, se eliminaron las dimensiones con entrada uno utilizando la función Squeeze[2] y se utilizó la forma resultante (B, T, CH) en las capas LSTM y lineales. Antes de la capa de salida, se regresó a la forma original (B, CH, T, 1) utilizando la función Unsqueeze[3], el modelo propuesto regresa la forma (B, N_CLASES, T, 1).

3.6. Optimización de hiperparámetros

Uno de los principales problemas al momento de entrenar un modelo de reconocimiento de actividades es seleccionar la mejor configuración de hiperparámetros. Esto se ha abordado utilizando diferentes técnicas como búsqueda en cuadrícula (GS), búsqueda aleatoria (RS) y la optimización de enjambre de partículas (PSO). Además, existe la forma manual que consta de realizar cambios manuales en los hiperparámetros y realizar pruebas.

Para reducir el tiempo que lleva encontrar la mejor configuración, se implementó la Optimización Bayesiana. Este método ha sido utilizado para la optimización de hiperparámetros en subconjuntos de big data [24]. La optimización se llevó a cabo utilizando las herramientas BoTorch [25] y Ax [26], debido a la facilidad que nos proporciona para realizar experimentos y la fácil integración que cuenta con Pytorch.

[2] Función Squeeze, https://pytorch.org/docs/stable/generated/torch.squeeze.html, último acceso 17/04/2023

[3] Función Unsqueeze, https://pytorch.org/docs/stable/generated/torch.unsqueeze.html, último acceso 17/04/2023

Tabla 7. Valor-F1 utilizando diferentes combinaciones de sensores utilizando datos del Subconjunto 1 y los sensores de ambas manos.

Combinación	C1	C2	C3	C4	C5	C6
# de canales	18	12	30	46	48	50
CNN-biLSTM (Sub 1)	**88.45%**	**89.86%**	**90.65%**	**89.32%**	**88.40%**	**89.41%**
DeepConvLstm	78.53%	84.70%	86.05%	81.53%	83.51%	85.23%

3.7. Evaluación

Para evaluar el desempeño de la arquitectura durante el entrenamiento, se utilizó la medida de exactitud de clasificación general multiclase de la biblioteca torchmetrics1, utilizando el optimizador Adam [27].

Al tratarse de un conjunto de datos desbalanceado, si el clasificador predice cada instancia como una clase mayoritaria y se utiliza la exactitud de clasificación general para evaluar el resultado, los resultados podrían lograr un alto rendimiento. Por lo tanto, la exactitud de clasificación general no es una medida apropiada para evaluar el modelo.

Por otro lado, el Valor-F1 (F1 score) toma en cuenta tanto los falsos positivos como los falsos negativos y muestra el equilibrio entre la precisión y recuperación. La precisión puede verse como $\mathrm{TP}/(\mathrm{TP}+\mathrm{FP})$ y la recuperación como $\mathrm{TP}/(\mathrm{TP}+\mathrm{FN})$ donde TP y FP son el número de verdaderos y falsos positivos. FP corresponde al número de falsos positivos. La fórmula del Valor-F1 está dada por:

$$\text{Valor-F1}= \sum_{i}^{\mathrm{N}} 2\times \mathrm{w}_i \frac{\text{precisión}_i .\ \text{recuperación}_i}{\text{precisión}_i + \text{recuperación}_i}, \tag{1}$$

donde $w_i = n_i/N$ es la proporción de muestra de la clase i, siendo n_i el número de muestras de la clase i-ésima y N el número total de muestras.

4. Resultados y discusión

Para evaluar el desempeño de la arquitectura propuesta se hizo una comparación contra DeepConvLSTM utilizando las configuraciones descritas por los autores en [23]. En la Tabla 5 se puede observar la lista de hiperparámetros. Se probó la arquitectura con diferentes combinaciones de sensores. En la Tabla 6 se pueden observar las combinaciones evaluadas. Para la evaluación se realizó un muestreo de los datos a 25 Hz.

Para obtener la mejor configuración de la arquitectura propuesta se utilizó optimización bayesiana, véase en la Fig. 3. Esta optimización se aplicó utilizando los datos de la configuración del Subconjunto 1, se realizaron 100 interacciones, los resultados se obtuvieron utilizando el Valor-F1.

Los mejores resultados se encontraron en la iteración 72 con un 90.84% de precisión. Los hiperparámetros obtenidos se pueden observar en la Fig. 4 (izquierda). Además, también se aplicó una optimización bayesiana a la arquitectura de referencia

Tabla 8. Valor-F1 utilizando datos de subconjunto 1 (Sub1) y subconjunto 2 (Sub2), así como los sensores de ambas manos aplicando filtro Kalman.

Combinación	C1	C2	C3	C4	C5	C6
# de canales	18	12	30	46	48	50
CNN-biLSTM (Sub 1)	89.60%	**90.71%**	90.67%	89.86%	87.88%	88.47%
CNN-biLSTM (Sub 2)	92.83%	**93.15%**	93.09%	92.90%	92.68%	93.12%

DeepConvLSTM, lo cual permitió identificar los hiperparámetros óptimos, que pueden observarse en la Fig. 4 (derecha).

Como se puede ver en la Tabla 7 la red propuesta CNN-biLSTM logra mejores resultados en todas las combinaciones evaluadas. Igualmente se puede notar que el mejor resultado obtenido es utilizando la combinación de sensores C3 (Acc, E4Acc, Gyro). Para reducir el ruido y las fluctuaciones registradas en las mediciones de los datos, se aplicó el filtro Kalman descrito en la sección 3.2.

Los resultados obtenidos por la arquitectura propuesta utilizando los datos del Subconjunto 1 se pueden observar en la Tabla 8. Se puede observar que el aplicar el filtro Kalman mejoró los resultados en las configuraciones donde se utilizan los datos del sensor Atr, sin embargo, se observó una disminución en el rendimiento al utilizar los datos del sensor E4, debido a que los datos de E4 tienen un rango de lectura más extenso y no estaban normalizados.

Los resultados obtenidos con la arquitectura propuesta utilizando los datos del Subconjunto 2 se pueden observar en la Tabla 8. Para el entrenamiento se utilizaron los hiperparámetros obtenidos en el Subconjunto 1. En comparación con el Subconjunto 1, el mejor resultado se obtuvo utilizando la combinación C2.

5. Conclusión y trabajo a futuro

En este artículo se propuso una nueva arquitectura de red convolucional profunda que combina capas convolucionales con LSTM para el reconocimiento de actividad humana en entornos industriales. Para probar la red se utilizó el conjunto de datos Openpack y se comparó contra una arquitectura de redes neuronales base. Se utilizó el Valor-F1 para comparar el desempeño. Finalmente, se logró un Valor-F1 de 93.12% con la configuración de Subconjunto 2.

Se realizaron diferentes experimentos para encontrar la mejor combinación de sensores. Además, también se exploró cómo el cambio de los hiperparámetros afecta el rendimiento del modelo, se aplicó la optimización bayesiana para obtener los mejores hiperparámetros. En comparación con los métodos propuestos en otros trabajos, CNN-biLSTM demostró un rendimiento superior cuando se trata de reconocer actividades en el área de la logística.

Si bien los resultados fueron buenos, solamente se realizaron experimentos utilizando los sensores IMU. En trabajos futuros se pretende adaptar la arquitectura propuesta para utilizar una combinación de los datos del sensor Kinect, visión profunda y la configuración actual, además de aplicar la normalización en los datos.

Agradecimientos. Este trabajo fue parcialmente financiado por Consejo Nacional de Ciencia y Tecnología (CONACYT) en México, con una beca al primer autor (1146285), así como por el Instituto Tecnológico de Sonora (ITSON) a través del programa PROFAPI.

Referencias

1. Ann, O. C., Theng, L. B.: Human activity recognition: A review. In: IEEE International Conference on Control System, Computing and Engineering, pp. 389-393 (2014) doi: 10.1109/iccsce.2014.7072750
2. Dinarevic, E. C., Husic, J. B., Barakovic, S.: Issues of human activity recognition in healthcare. In: 18th International Symposium INFOTEH-JAHORINA, pp. 1–6 (2019) doi: 10.1109/infoteh.2019.8717749
3. Malaise, A., Maurice, P., Colas, F., Ivaldi, S.: Activity recognition for ergonomics assessment of industrial tasks with automatic feature selection. IEEE Robotics and Automation Letters, vol. 4, no. 2, pp. 1132–1139 (2019) doi: 10.1109/lra.2019.2894389
4. Gkournelos, C., Karagiannis, P., Kousi, N., Michalos, G., Koukas, S., Makris, S.: Application of wearable devices for supporting operators in human-robot cooperative assembly tasks. Procedia CIRP, vol. 76, pp. 177–182 (2018) doi: 10.1016/j.procir.2018.01.019
5. Ignatov, A.: Real-time human activity recognition from accelerometer data using convolutional neural networks. Applied Soft Computing, vol. 62, pp. 915–922 (2018) doi: 10.1016/j.asoc.2017.09.027
6. Cust, E. E., Sweeting, A. J., Ball, K., Robertson, S.: Machine and deep learning for sport-specific movement recognition: A systematic review of model development and performance. Journal of Sports Sciences, vol. 37, no. 5, pp. 568–600 (2018) doi: 10.1080/02640414.2018.1521769
7. Chen, L., Wei, H., Ferryman, J.: A survey of human motion analysis using depth imagery. Pattern Recognition Letters, vol. 34, no. 15, pp. 1995–2006 (2013) doi: 10.1016/j.patrec.2013.02.006.
8. Koskimaki, H., Huikari, V., Siirtola, P., Laurinen, P., Roning, J.: Activity recognition using a wrist-worn inertial measurement unit: A case study for industrial assembly lines. In: 2009 17th Mediterranean Conference on Control and Automation, pp. 401–405 (2009) doi: 10.1109/med.2009.5164574
9. Niemann, F., Reining, C., Rueda, F. M., Nair, N. R., Steffens, J. A., Fink, G. A., Hompel, M. T.: LARa: Creating a dataset for human activity recognition in logistics using semantic attributes. Sensors, vol. 20, no. 15 (2020) doi: 10.3390/s20154083
10. Yoshimura, N., Morales, J., Maekawa, T., Hara, T.: OpenPack: A large-scale dataset for recognizing packaging works in IoT-enabled logistic environments (2022) doi: 10.48550/ARXIV.2212.11152
11. Dallel, M., Havard, V., Baudry, D., Savatier, X.: InHARD-Industrial human action recognition dataset in the context of industrial collaborative robotics. In: 2020 IEEE International Conference on Human-Machine Systems, pp. 1–6 (2020) doi: 10.1109/ICHMS49158.2020.9209531
12. Feldhorst, S., Masoudenijad, M., Ten-Hompel, M., Fink, G., A.: Motion classification for analyzing the order picking process using mobile sensors – General concepts, case studies and empirical evaluation. In: Proceedings of the 5th International Conference on Pattern Recognition Applications and Methods, vol. 1, pp. 706–713 (2016) doi: 10.5220/0005828407060713
13. Grzeszick, R., Lenk, J. M., Rueda, F. M., Fink, G. A., Feldhorst, S., Ten-Hompel, M.: Deep neural network based human activity recognition for the order picking process. In:

Proceedings of the 4th International Workshop on Sensor-based Activity Recognition and Interaction, no. 14, pp. 1–6 (2017) doi: 10.1145/3134230.3134231
14. Rahman, M. A., Mia, Y., Rahman-Masum, R., Hasan-Abid D. M., Islam, T.: Real time human activity recognition from accelerometer data using convolutional neural networks. In: 2022 7th International Conference on Communication and Electronics Systems, pp. 1394–1397 (2022) doi: 10.1109/ICCES54183.2022.9835797
15. Panwar, M., Ram Dyuthi, S., Chandra Prakash, K., Biswas, D., Acharyya, A., Maharatna, K., Gautam, A., Naik, G. R.: CNN based approach for activity recognition using a wrist-worn accelerometer. In: 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, pp. 2438–2441 (2017) doi: 10.1109/EMBC.2017.8037349
16. Roggen, D., Calatroni, A., Rossi, M., Holleczek, T., Forster, K., Troster, G., Lukowicz, P., Bannach, D., Pirkl, G., Ferscha, A., Doppler, J., Holzmann, C., Kurz, M., Holl, G., Chavarriaga, R., Sagha, H., Bayati, H., Creatura, M., Millan, J. R.: Collecting complex activity datasets in highly rich networked sensor environments. In: Seventh International Conference on Networked Sensing Systems, pp. 233–240 (2010) doi: 10.1109/INSS.2010.5573462
17. Chavarriaga, R., Sagha, H., Calatroni, A., Digumarti, S. T., Tröster, G., Millán, J. del R., Roggen, D.: The Opportunity challenge: A benchmark database for on-body sensor-based activity recognition. Pattern Recognition Letters, vol. 34, no. 15, pp. 2033–2042 (2013) doi: 10.1016/j.patrec.2012.12.014
18. Reiss, A., Stricker, D.: Introducing a new benchmarked dataset for activity monitoring. In: 16th International Symposium on Wearable Computers, pp. 108–109 (2012) doi: 10.1109/ISWC.2012.13
19. Moya-Rueda, F., Grzeszick, R., Fink, G., Feldhorst, S., Hompel, M.: Convolutional neural networks for human activity recognition using body-worn sensors. Informatics, vol. 5, no. 2, pp. 1–17 (2018) doi: 10.3390/informatics5020026
20. Rippel, O., Snoek, J., Adams, R., P.: Spectral representations for convolutional neural networks. In: Proceedings of the 28th International Conference on Neural Information Processing Systems, vol. 2 (2015) doi: 10.48550/arXiv.1506.03767
21. Yu, Y., Si, X., Hu, C., Zhang, J.: A review of recurrent neural networks: LSTM cells and network architectures. Neural Computation, vol. 31, no. 7, pp. 1235–1270 (2019) doi: 10.1162/neco_a_01199
22. Xia, K., Huang, J., Wang, H.: LSTM-CNN architecture for human activity recognition. IEEE Access, vol. 8, pp. 56855–56866 (2020) doi: 10.1109/ACCESS.2020.2982225
23. Ordóñez, F. J., Roggen, D.: Deep convolutional and LSTM recurrent neural networks for multimodal wearable activity recognition. Sensors, vol. 16, no. 1, pp. 2–25 (2016) doi: 10.3390/s16010115
24. Klein, A., Falkner, S., Bartels, S., Hennig, P., Hutter, F.: Fast bayesian optimization of machine learning hyperparameters on large datasets. Artificial intelligence and statistics, pp. 528–536 (2017) doi: 10.48550/arXiv.1605.07079
25. Balandat, M., Karrer, B., Jiang, D. R., Daulton, S., Letham, B., Wilson, A. G., Bakshy, E.: BOTORCH: A framework for efficient monte-carlo bayesian optimization (2020) doi: 10.48550/arXiv.1910.06403
26. Baird, S. G., Liu, M., Sparks, T. D.: High-dimensional bayesian optimization of hyperparameters for an attention-based network to predict materials property: a case study on CrabNet using Ax and SAASBO. Computational Materials Science, vol. 211 (2022) doi: 10.1016/j.commatsci.2022.111505