Universidad Nacional de Ciudad Ho Chi Minh
UNIVERSIDAD DE CIENCIAS NATURALES
FACULTAD DE TECNOLOGÍA DE LA INFORMACIÓN

Tema: Detección y zonificación de objetos con comportamiento extraño en el apartamento con cámaras CCTV

Instructor: Prof. Asociado Dr. Le Hoang Thai

Grupo de estudiantes: • Do
Vuong Phuc – 19127242 • Vo
Hoang Bao Duy – 19127027 • Le
Minh Si – 19127064

Ciudad Ho Chi Minh, 2022

TABLA DE

TABLA DE CONTENIDOS

1 Información del equipo y desglose del trabajo 1.1 Información del equipo

• Estudiante 1: Do Vuong Phuc
o MSSV: 19127242 o
Correo electrónico: phuc16102001@gmail.com o
Teléfono de contacto: (+84) 707 953 475 •
Estudiante 2: Vo Hoang Bao Duy o MSSV: 19127027 o
Correo electrónico: v.hbaoduy@gmail.com .com
o Teléfono de contacto: (+84) 776 562 199 •
Estudiante 3: Le Minh Si o MSSV: 19127064 o
Dirección de correo electrónico: hungtiensi.lms@gmail.com
o Teléfono de contacto: (+84) 842 429 138 •
Instructor: Asoc. Dr. Le Hoang Thai o Agencia de
trabajo: Facultad de Tecnología de la Información,
Universidad de Ciencias Naturales o Dirección de correo
electrónico: lhthai@hcmus.edu.vn

1.2 Tabla de desglose de puestos

Código SV.	Nombre y apellido	Trabajo	nivel de finalización
19127242	Do Vuong Phuc	- Lluvia de ideas - Escribir la parte inicial - Demostraciones de diseño - Aprende y escribe sobre MediaPipe - Revisar y formatear documentos.	100%
19127027	Vo Hoang Bao Duy	- Escribir una descripción general del problema. - Aprenda LSTM y RNN - Escribir propuesta de modelo. - Escribir identificación y rastreo.	100%
19127067	Le Minh Si	- Aprende y escribe sobre CNN - Escribir dirección de desarrollo - Buscar en la base de datos - Diapositivas de diseño	100%

2 Lista de siglas

Acrónimos	Palabra original / Inglés
circuito cerrado de televisión	Circuito Cerrado de Televisión y Video
AI	Inteligencia Artificial
LSTM	Memoria a corto plazo
CNN	Red neuronal de convolución
RNN	Red neuronal recurrente
CV	Visión por computador
DL	Aprendizaje profundo
ANA	Red neuronal artificial
PNL	Procesamiento natural del lenguaje
yolo	Miras solo una vez
CONTRA	Seguimiento de múltiples objetos
CLASIFICAR	Seguimiento simple de objetos en línea en tiempo real

3 Introducción 3.1

Razones para elegir el tema En

la era de la tecnología de la información y el desarrollo técnico, el volumen de datos y la cantidad de información que proviene de muchas fuentes aumenta constantemente, especialmente una de ellas es la cantidad de imágenes, fotos, videos, etc. .. cada vez más y más grande. Por lo tanto, la clasificación y detección de problemas a partir de las imágenes y videos recopilados es una necesidad sumamente necesaria al servicio de la investigación y desarrollo de aplicaciones que ayuden a las personas a resolver problemas y dificultades en la vida cotidiana.

En los últimos años, con el gran desarrollo del campo de la Visión por Computador. Grandes sistemas de procesamiento de imágenes en el mundo como Facebook, Google... han puesto sus productos en aplicaciones prácticas como autos sin conductor, reconocimiento facial de usuarios, etc.

En muchos países, los modelos de uso de CCTV para monitoreo han logrado muchos resultados positivos. Como proyectos que utilizan cámaras para el reconocimiento temprano y la prevención del suicidio [3], específicamente en el río Han en Corea [4]

Los métodos actuales han logrado un éxito seguro, como la detección de comportamientos anormales en hogares inteligentes [5], en multitudes [6]. Sin embargo, todavía no hay solicitudes para edificios de apartamentos y edificios corporativos.

Actualmente, la mayoría de los apartamentos están equipados con sistemas de cámaras de vigilancia, que proporcionan datos de imágenes en cualquier momento y en cualquier lugar. Sin embargo, mientras que la cantidad de cámaras implementadas parece crecer a un ritmo sorprendente, así como con una calidad de imagen mejorada,

y el objetivo principal es el monitoreo en vivo o el registro de datos. Con un número tan elevado de cámaras, es necesario disponer de un equipo de guardias naturales que vigilen de cerca y de forma continuada para garantizar la situación de seguridad de la zona así como conseguir la máxima eficacia.

Entonces, con una serie de mejoras en la tecnología que permite que los sistemas de cámaras detecten y alerten al equipo de vigilancia cuando se detecta alguna actividad sospechosa. Con el desarrollo de inteligencia artificial (IA), aprendizaje automático, aprendizaje profundo; Las actividades sospechosas e inusuales se entienden y detectan automáticamente analizando el comportamiento de las personas en sus acciones, entornos o eventos específicos en los que están participando.

3.2 Propósito de la investigación

Hoy en día, todos los apartamentos/edificios de las grandes ciudades están equipados con un sistema de cámaras de seguridad en los pasillos. Por lo tanto, la observación de las cámaras de seguridad del equipo de vigilancia a veces comete errores, con solo 1 segundo sin observar actividades sospechosas (robo, robo, ...) que pueden afectar la seguridad del edificio de apartamentos. Por lo tanto, para apoyar al equipo de monitoreo y garantizar que las cámaras en el apartamento estén siempre monitoreadas las 24 horas del día, los 7 días de la semana, se requiere la intervención de tecnología. Específicamente, en este caso, es necesario tener un modelo para extraer características de video para poder detectar y localizar con precisión objetos con comportamiento anormal en tiempo real.

Con el desarrollo de modelos de aprendizaje profundo (Deep learning), junto con los problemas de hardware (CPU, GPU, ...) no se convierten en una preocupación en el entrenamiento de modelos de aprendizaje profundo. Por lo tanto, elegir un modelo de aprendizaje profundo en este caso es un enfoque que puede resolver el problema.

3.3 Objetivos de la investigación

Como se analizó anteriormente, detectar y zonificar automáticamente el comportamiento anómalo de la cámara es una necesidad urgente para garantizar la seguridad, identificando los riesgos más tempranos para que puedan prevenirse. Con el propósito de que la investigación resuelva los problemas antes mencionados, en este estudio nos planteamos los siguientes objetivos de investigación.

El objetivo de la investigación es detectar y localizar objetos con comportamiento extraño en tiempo real a través de video extraído de la cámara.

Para lograr el objetivo anterior, necesitamos aprender y estudiar los problemas de los modelos de aprendizaje profundo (Deep Learning), a saber: CNN extrayendo características de video, rastreando objetos, estimando Pose Estimation y combinado con LSTM para clasificar el comportamiento, resolviendo así el problema planteado .

3.4 Objeto de investigación y ámbito de investigación

3.4.1 Objeto de investigación Para poder

cumplir con el objetivo de investigación, primero debemos estudiar los temas relacionados con el procesamiento de video (formatos, estándares de tipo de película) para tener los pasos de implementación adecuados.
A continuación, necesitamos explorar estudios relacionados sobre detección de objetos, seguimiento de objetos y estimación de poses en video.

Además, el estudio también examina, analiza y evalúa las ventajas y desventajas de los métodos de detección, seguimiento y predicción de la postura del sujeto, combinados con modelos computacionales de los métodos disponibles, para construir un nuevo modelo adecuado para el objetivo del problema al detectar y zonificación de objetos con comportamiento extraño.

3.4.2 Alcance de la investigación El

tema se realiza principalmente en detectar, rastrear y predecir la postura del sujeto, zonificando así los objetos con comportamiento extraño a través de cámaras instaladas en edificios, apartamentos. El conjunto de datos para la implementación son videos estándar para la investigación en campos relacionados y datos recopilados en la práctica de cámaras de video ubicadas en varios edificios y apartamentos en Vietnam.

El modelo propuesto tendrá una expectativa adecuada para el problema de detección y localización de objetos con comportamiento extraño, el conjunto de datos para entrenar este modelo es de tamaño suficiente para entrenar el modelo. Además, el modelo también puede ejecutarse en sistemas de hardware menos potentes en el proceso de análisis de videoclips de la cámara.

3.5 Métodos de investigación Aplicar

muchos métodos diferentes: encuesta, experimento, análisis y modelado.

Específicamente: • Estudiar, analizar y evaluar los métodos existentes. Lea artículos relacionados con el problema de investigación, evalúe los pros y los contras, y luego elija el método adecuado para resolver los requisitos planteados.

• Construir modelos sobre datos reales (si es posible) para mejorar y dar al modelo la mejor viabilidad y resultados.

Métodos de aprendizaje teórico: comprensión de los modelos de aprendizaje profundo: CNN, LSTM (RNN) y estimación de pose por MediaPipe.

Método experimental sobre datos de muestra: estudio y uso de modelos CNN, LSTM y RNN para construir, instalar y entrenar el modelo propuesto. A partir del modelo propuesto entrenado, evalúe los resultados y el rendimiento del modelo en el conjunto de datos independientemente de los datos de entrenamiento.

3.6 Temas que contribuyen

• Estructura de la red CNN aplicada al problema de detección (detecting) y seguimiento (tracking) para el problema dado. • Se combina el modelo MediaPipe de BlazePose y GHUM • Se aplica la estructura LSTM para estimar la figura a partir de la cual clasificar los comportamientos.

micro anormal.

• En este informe, presentaremos un modelo de combinación entre CNN y LSTM para puede resolver el problema planteado.

• Recolectar y construir un sistema de datos que sirva para el entrenamiento y evaluación del modelo propuesto.

3.7 Contenido del informe

El contenido del informe incluye las siguientes partes principales:

- Descripción general del problema de detección de comportamientos extraños y el problema de zonificación de objetos: En esta parte, el informe se centra en introducir y generalizar el problema de detección y seguimiento de objetos. y estimación de la forma del cuerpo para clasificar comportamientos extraños. Además, presentar artículos e investigaciones relacionadas con el problema que estamos investigando.

- Antecedentes teóricos: En esta sección, nos enfocamos en la base teórica al momento de implementar el tema, específicamente temas relacionados con CNN, RNN, LSTM, BlazePose y modelos modelo. GHUM.
- El método de evaluación del modelo propuesto. •
Construyendo un modelo de aprendizaje profundo para detectar y localizar objetos con comportamiento extraño en el apartamento: En esta sección presentamos el modelo que hemos investigado y propuesto.
- Experimentar y evaluar resultados: Nos enfocamos en realizar análisis de conjuntos de datos para entrenar el modelo. En base a eso, se ha evaluado la precisión del modelo. • Conclusión y dirección de desarrollo.

4 Descripción general del problema de detección de comportamientos extraños y rastreo del área para símbolo

4.1 El problema de detectar comportamientos extraños

El problema de detectar un comportamiento extraño se ha estudiado durante mucho tiempo [1], el propósito del problema es determinar si el objeto específico aquí es la persona en la imagen o los cuadros de video tienen un comportamiento anormal o no. Para resolver este problema, necesitamos recopilar datos, procesar los datos, extraer características y entrenar el modelo para realizar la determinación de si cada cuadro del video tiene algún comportamiento considerado inusual.

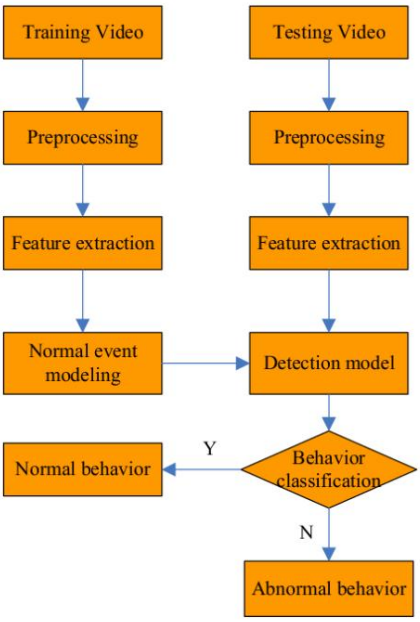

Figura 1. Modelo básico de detección de comportamiento anómalo

El problema de detectar un comportamiento anormal puede considerarse como un problema de clasificación.

Clasificación: es un proceso de procesamiento para ordenar y asignar etiquetas (etiquetas) de objetos en una determinada clase (clase). Las muestras de datos u objetos se clasifican en función de sus propiedades o características. Específicamente en el problema mencionado, necesitamos usar las características de cada cuadro de video para considerar a qué clase pertenecen los objetos en el video: comportamiento normal o comportamiento anormal.

Con el desarrollo de la tecnología de la información, la explosión digital y de datos como la actual, la clasificación se ha convertido en una necesidad indispensable en muchos campos, y el problema que estamos considerando no es una excepción. Actualmente, existen muchos modelos de clasificación diferentes, por lo que es muy importante elegir un modelo de clasificación que brinde eficiencia y precisión.

En general, la mayoría de los comportamientos humanos se pueden identificar por la forma del cuerpo. Por lo tanto, usamos la estimación de pose para ayudar a que el problema de clasificación funcione mejor [2] .

Estimación de pose: es un problema común en el campo de la Visión por Computador (CV). Su propósito es determinar la posición y dirección de una persona u objeto. Por lo general, esto es predecir las posiciones de puntos clave como manos, cabeza, pies, etc. en el caso de Estimación de pose humana .

Figura 2. Estimación de pose

4.2 Problema de rastreo y zonificación de objetos

El seguimiento de objetos es también uno de los problemas clásicos de Computer Vision (CV). Aquí, el problema tiene que crear un conjunto de objetos detectados (Detección de objetos), luego asignar un identificador (ID) a cada objeto detectado y finalmente rastrear los objetos cuando se detectan y pasar a los siguientes cuadros (frames) en el video.
El seguimiento de personas en video es un caso específico de problema de seguimiento de objetos.

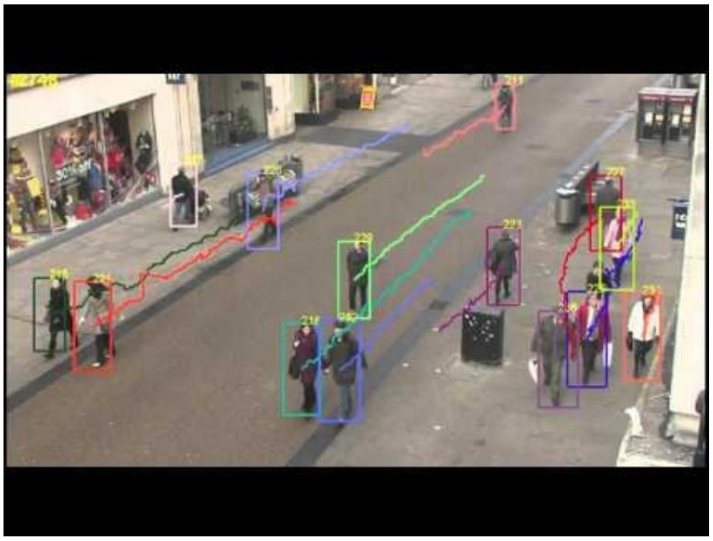

Figura 3. Seguimiento de objetos

Con base en el rastreo de objetos, a partir de ese momento, se realiza la zonificación de objetos. Con la efectividad de los modelos de aprendizaje profundo como CNN, su uso en el problema de rastreo es muy factible [7].

4.3 Situación de la investigación

En los últimos años, el problema de detectar comportamientos anormales ha sido estudiado activamente por grupos que utilizan diferentes métodos. Estos son algunos de los métodos que encontramos.

En este artículo [8], los autores proponen la identificación de actividades y comportamientos anormales utilizando modelos RNN como: LSTM, GRU... En la identificación de actividades se considera como etiquetado para el etiquetado de secuencias, en el que los comportamientos anormales se marcan en función de su desviación de los comportamientos normales.

Los autores [9] combinan el rastreo de objetos y la extracción de características utilizando el modelo CNN y el uso del modelo LSTM para proporcionar una clasificación de comportamientos con una eficiencia relativamente significativa en el problema.

5 Fundamentos teóricos

5.1 Red neuronal de convolución (CNN)

En los modelos de aprendizaje profundo, CNN, o red neuronal convolucional, puede considerarse como uno de los modelos destacados en DL. En los últimos años, el modelo CNN ha sido ampliamente utilizado en el campo CV, construido con la tarea de reconocer y clasificar imágenes. Entre ellos, la detección de objetos es uno de los campos de amplia aplicación.

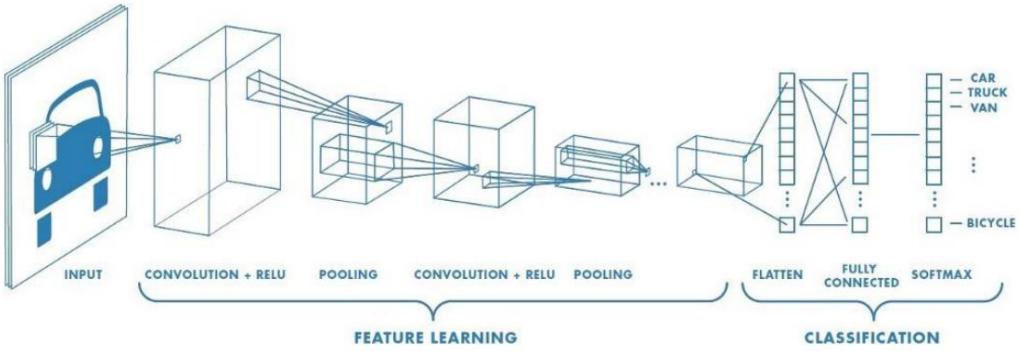

Figura 4. Modelo básico de CNN en tarea de clasificación de imágenes

En este modelo, en lugar de usar solo capas completamente conectadas como en ANN, el modelo CNN también usa capas de convolución y capas de agrupación antes de usar completamente conectadas para clasificación o predicción.

5.1.1 Capa de Convolución – Capa de Convolución.

La capa de acumulación es la primera en extraer características de las imágenes de entrada en el modelo CNN. El uso de estas capas convolucionales es mantener la relación entre los píxeles (píxeles) usando el kernel (inicializado) y deslizarlos sobre los píxeles.

La convolución es una operación matemática realizada con dos F y $_{gramo}$, El resultado será una función . En funciones numéricas: el resultado de la convolución fg también es muy importante en el procesamiento de imágenes, la convolución es ampliamente utilizada para extraer características de una imagen.

La fórmula de convolución entre la función $fxy(,)$ y filtros $kxy(,)$ (con tamaño min imagen) se expresa de la siguiente manera:

$$fxy(,) = \sum_{en=-\frac{Minnesota}{2}}^{\frac{Minnesota}{2}} \sum_{en=-\frac{Minnesota}{2}}^{\frac{Minnesota}{2}} ( ) kuvfxuyu ( ) ,$$

Ejemplo: dada una entrada de tamaño 5x5 y el filtro inicializado para tener un tamaño de 3x3. También La implementación de la convolución se describe a continuación.

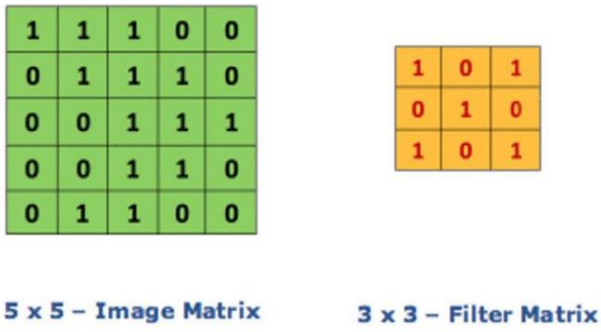

Figura 5. Convolución entre imagen de entrada y filtro

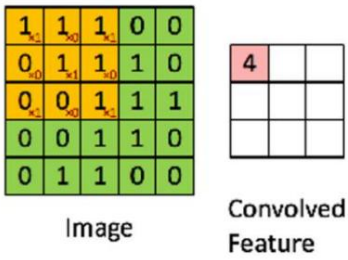

Figura 6. Simulación del proceso de filtro de diapositivas en la imagen (1)

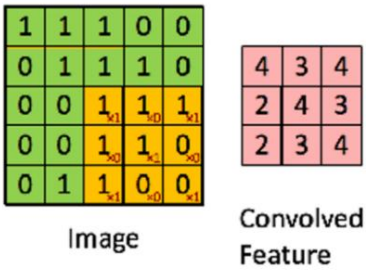

Figura 7. Simulación del proceso de filtro deslizante sobre la imagen (2)

Filtro: es una parte importante de la capa de convolución. Esta es una matriz inicializada con tamaño mx n, y el tamaño es muy pequeño en comparación con el tamaño de la imagen (generalmente inicializada en tamaño 5x5 o 3x3) para obtener mejores resultados.

Stride: en el proceso de deslizar el filtro sobre la imagen de entrada, el filtro se desplazará horizontal o verticalmente un valor a su vez, este valor se denomina Stride (paso deslizante). En cada turno (deslice de izquierda a derecha, de arriba a abajo), calculará el resultado para el píxel en consideración mediante la fórmula de acumulación descrita anteriormente.

Relleno: a veces, en la implementación de convolución, el filtro de filtro no es adecuado para la imagen de entrada. Podemos agregar relleno (borde) a los 4 bordes de la imagen antes de realizar la convolución para garantizar que el tamaño de salida sea constante.

Mapa de características: es el resultado después de realizar la convolución entre la imagen de entrada y el filtro al escanear todo el recorrido. El tamaño de la salida se calcula mediante la siguiente fórmula:

$$EN_1 = \frac{W_0 PF2 + - 1}{S} + \quad , H_1 = \frac{HP_0 F - 2 - 1 +}{S}$$

Con

- $W_0 H_{,0}$ es el tamaño de la imagen de entrada.
- $W_1 H_{,1}$ es el tamaño de la imagen de entrada.
- ^{PAG} es el tamaño del relleno.

- F es el tamaño del filtro. es el
- S paso de deslizamiento.

Usando la imagen de entrada con el tamaño WH$_0$ $_0$ siendo el número de canales de la imagen), la convolución se realiza en cada canal a su vez y luego se suma entre los canales.
Podemos agregar sesgo al resultado después de realizar la convolución.

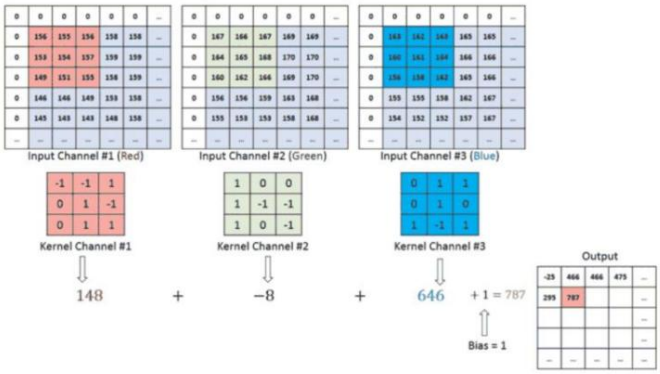

Figura 8. El proceso de convolución con imágenes multicanal y aplicación de sesgo

5.1.2 Capa de agrupación.

En el modelo CNN, generalmente entre las clases de acumulación, las personas suelen insertar una capa de agrupación para reducir la cantidad de parámetros si la imagen de entrada es demasiado grande. La aplicación de una capa de agrupación reduce el tamaño del espacio de muestra, pero conserva las características básicas de la imagen de entrada. En este proceso, también utilizando el tamaño de la ventana deslizante en la imagen, la agrupación tiene muchos tipos diferentes, tales como:

- Agrupación máxima: para cada diapositiva de ventana, elija el píxel más grande.
- Agrupación media: selecciona la media entre píxeles correspondiente al tamaño de la ventana.
- Sum pooling: la suma de todos los píxeles de la ventana.

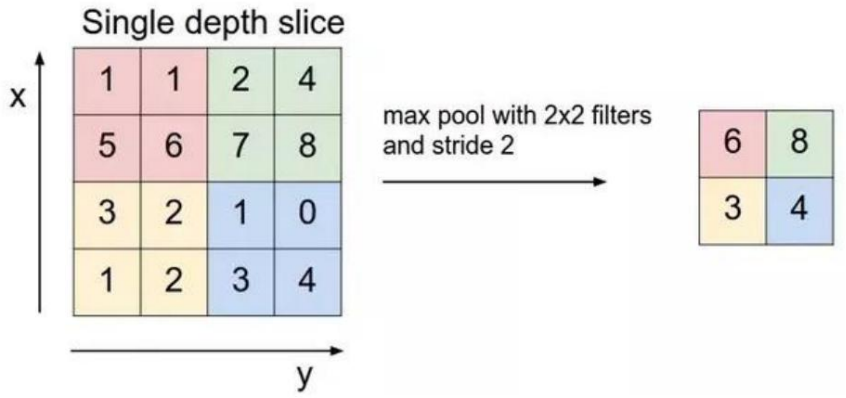

Figura 9. Proceso de aplicación de la capa Max Pooling después de la convolución

5.1.3 Función de activación – Función de activación.

Normalmente, después de realizar la convolución, al mapa de características calculado se le aplicará la función de activación a todos los valores del mapa de características. Algunas funciones de activación comúnmente utilizadas son: ReLU (Unidad lineal rectificada), Tanh, Sigmoid.

• ReLU: es una función descrita por la fórmula:

$$( ) = \max(0,)$$

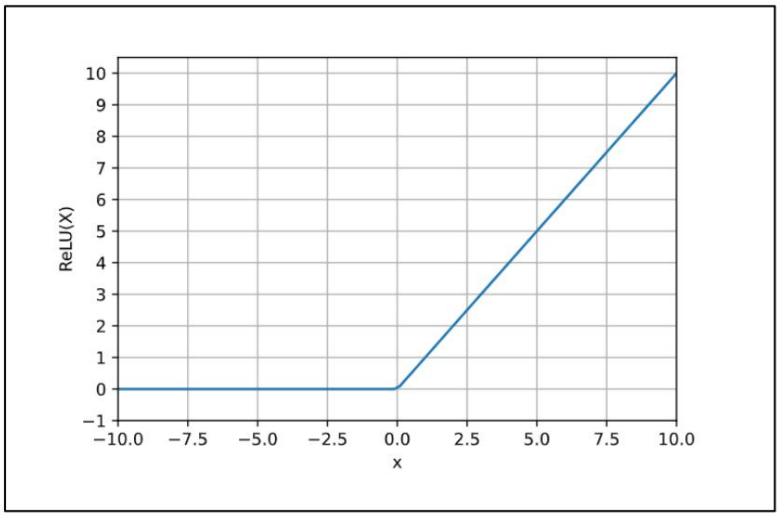

Figura 10. Función de Activación - ReLU

• Tanh: es una función de activación no lineal definida por la fórmula:

$$\tanh( ) = \frac{- \quad -}{+ -}$$

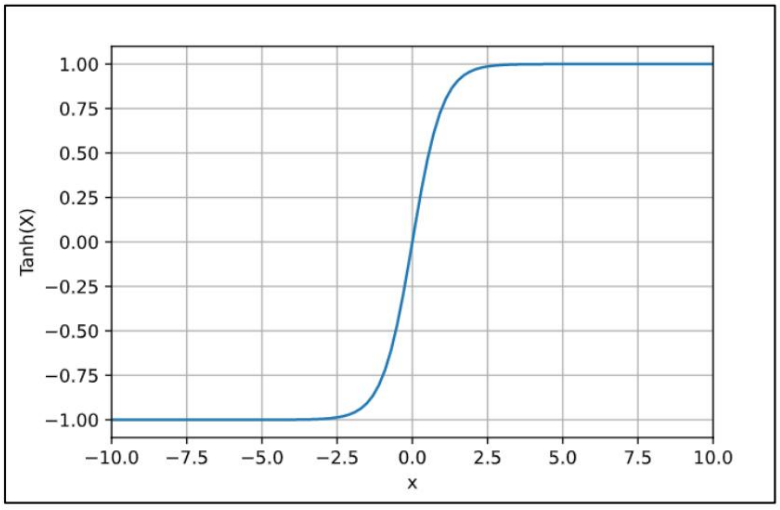

Figura 11. Función de disparo - Tanh

• Sigmoide: es una función de activación no lineal.

$$f( ) = \frac{1}{1 + -}$$

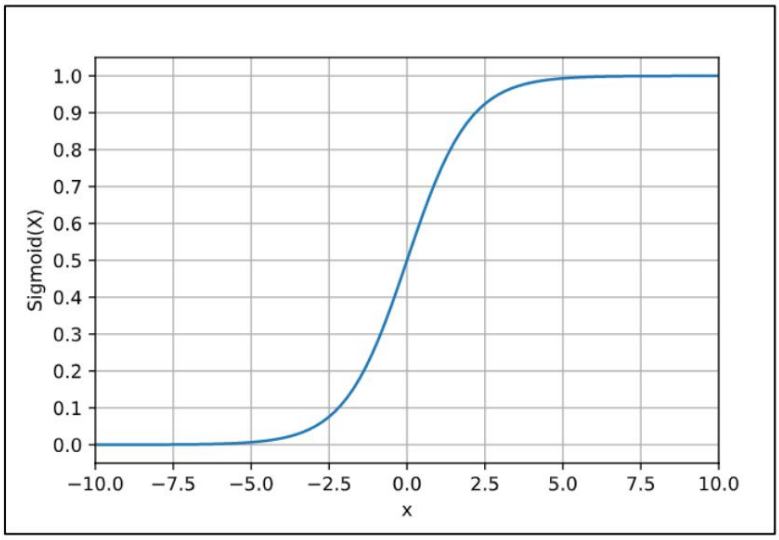

Figura 12. Función de activación - Sigmoid

5.1.4 Capa completamente conectada.

Después de realizar la extracción de características de la imagen a través de capas de convolución y agrupación, luego para realizar el proceso de predicción de resultados, usamos la capa Completamente conectada agregada después de una red CNN. Esta capa es similar a las capas del modelo ANN.

5.2 Red neuronal recurrente (RNN)

El modelo RNN también es una forma de modelo ANN, el modelo se usa a menudo para datos significativos secuenciales (o los llamados datos secuenciales) como: voz (habla), texto, sonido..., es decir, si cambia el secuencia de los datos, el problema obtendrá un resultado diferente.

RNN nos ayuda a guardar el estado o información de la entrada anterior para crear una secuencia de aprendizaje. Aquí está la arquitectura común del modelo RNN:

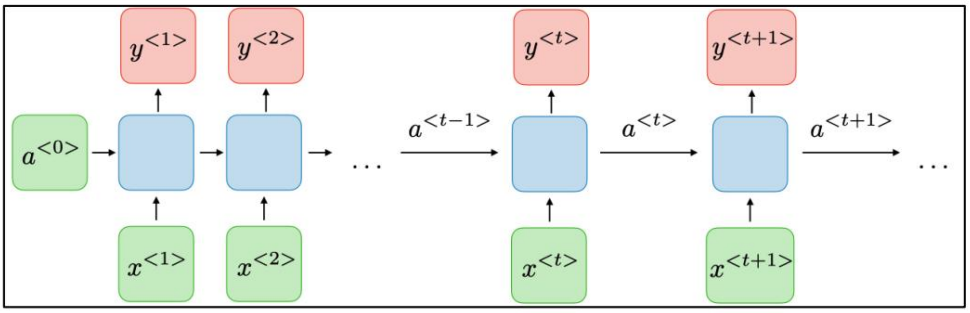

Figura 13. Arquitectura común del modelo RNN.

en cada paso t, tenemos el valor de la función de activación aplicada a^{t} y el valor de salida es y^{t}. Podemos realizarlo así:

$$a^{t} = g(W_{a}(x^{t-1}, a^{segundo}) + b_{a})$$

$$y^{t} = g(W_{y}a^{t} + b_{y})$$

Con :

- $W_{aa}, W_{ax}, W_{ya}, b_a, b_y$ son los coeficientes temporalmente compartidos
- g_1, g_2 son funciones de activacion

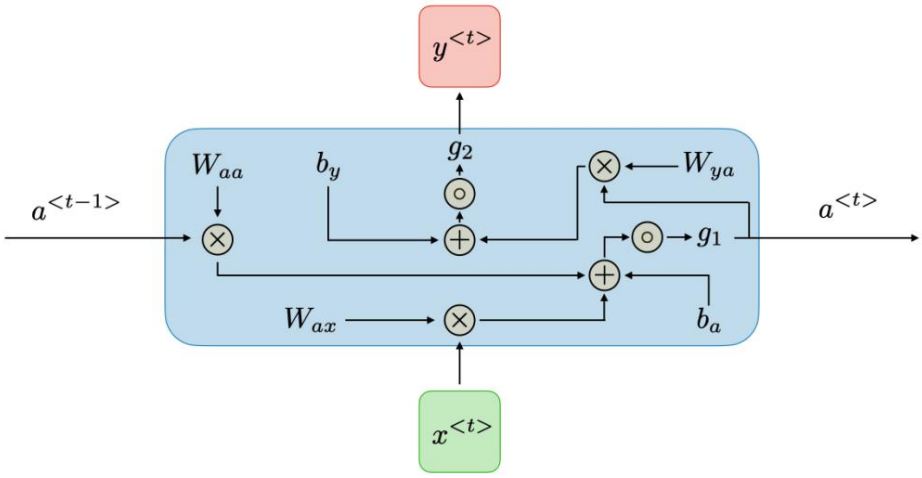

Figura 14. Modelo computacional de RNN

Los modelos RNN se utilizan en muchas áreas del procesamiento del lenguaje natural (NLP) y el reconocimiento de voz. Para cada aplicación del modelo RNN a un propósito específico, los RNN se dividen en las siguientes categorías:

Llamar T_x es el número de datos de entrada, T_y es el valor a devolver.

- Uno a uno: red neuronal tradicional. $T_x = T_y = 1$.

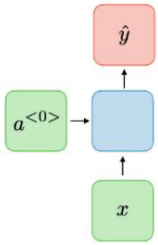

Figura 15. Modelo uno a uno

- Uno a muchos: comúnmente utilizado en la generación de música. $T_x = 1, T_y > 1$.

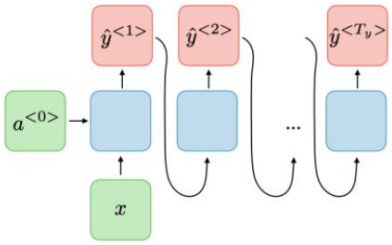

Figura 16. Modelo de uno a muchos

- Many-to-one: se utiliza en todas las clasificaciones emocionales (Clasificación Sentimiento). $T_x > 1$ años $T_y = 1$.

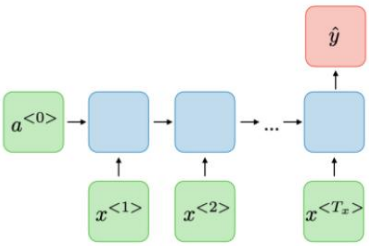

Figura 17. Modelo muchos a uno

- Muchos a muchos: a menudo se usa en el campo de la traducción automática (Machine si si $T_x T_y$ Translation), o se usa en el reconocimiento de entidades de nombre.

  $T_x T_y =$.

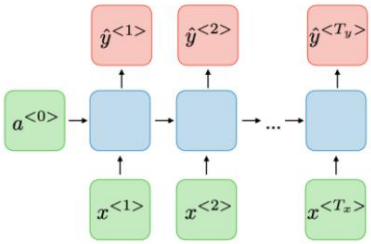

Figura 18. Modelo muchos a muchos (1)

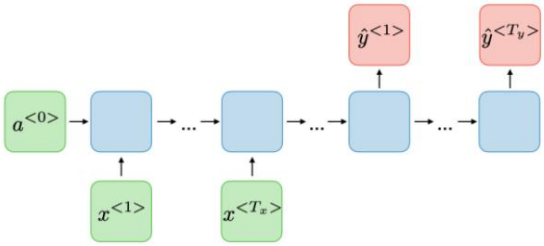

Figura 19. Modelo muchos a muchos (2)

Sin embargo, en el modelo RNN anterior, hay dos grandes problemas en el proceso de entrenamiento:

- Explosión de gradientes: en el proceso de retropropagación (Backpropagation) puede dar lugar a derivados muy grandes en cada parámetro, lo que lleva a una actualización incorrecta de los parámetros, esto se mejora mediante el uso de la técnica de reducción de precios.

- Gradientes de fuga: los valores derivados son demasiado pequeños (aproximadamente cero), por lo que la actualización de los parámetros durante el entrenamiento no tiene sentido. Este problema se puede resolver usando LSTM.

5.3 Memoria a corto plazo (LSTM)

Para resolver el problema que enfrenta el modelo RNN, Sepp Hochreiter y Juergen Schmidhuber concibieron LSTM para evitar el problema de los gradientes de fuga. LSTM es un modelo extendido de RNN, que es esencialmente más expansión de memoria que el modelo RNN convencional.

LSTM permite que RNN recuerde entradas anteriores durante largos períodos de tiempo, porque LSTM mantiene la misma información en la memoria que la memoria de la computadora. El LSTM puede leer, escribir y borrar información de la memoria. En LSTM, incluye puertas para regular mejor el flujo de información a través de las unidades. Específicamente:

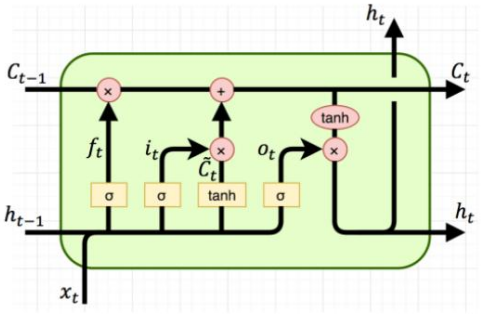

Figura 20. Modelo básico de LSTM

Podemos expresar la fórmula de LSTM de la siguiente manera:

$$f_t = \sigma_{gt}(W_f x_t + U_f h_{t-1} + b_f)$$
$$i_t = \sigma_{que}(W_i x_t + U_i h_{t-1} + b_i)$$
$$o_t = \sigma_{\_}(W_0 x_t + U_0 h_{t-1} + b_0)$$
$$\tilde{c}_t = \sigma_c(W_c x_t + U_c h_{t-1} + b_c)$$
$$c_t = f_t \circ c_{t-1} + i_t \circ \tilde{c}_t$$
$$h_t = o_t \circ \sigma_h(c_t)$$

Con:

- $\sigma_{\_}$ es la función de activación sigmoidea
- σ_c es la función de activación tanh
- σ_h es la función de activación tanh o la $\sigma_h(x) = x$
- F_i puerta de olvido es el puerto de entrada
- i_t es el puerto de salida es el puerto de
- o_t entrada de la celda es el estado de la
- $\tilde{c}_t$ celda es el vector del estado oculto
- c_t
- h_t

Al observar la estructura de una unidad de LSTM, se puede ver que las puertas de salida usan la función de activación sigmoidea (el valor de esta función está entre 0 y 1), por lo que se resolvió el problema de los gradientes de fuga, ayudando al proceso de entrenamiento . El modelo converge rápidamente y brinda mayor precisión que el modelo RNN tradicional.

6 Construcción de un modelo para detectar y localizar objetos con comportamiento extraño en

En esta sección ,

presentaremos los modelos utilizados en cada tarea por separado, implementando así una combinación de modelos para resolver el problema planteado anteriormente. Incluye: • Técnicas de detección de objetos • Rastreo de objetos detectados • Extracción de las figuras del

objeto de rastreo • Modelo combinado de aprendizaje profundo para resolver el problema planteado.

6.1 Detección y seguimiento de objetos

Detección de objetos En la tarea de detectar y

rastrear objetos (específicamente, personas) en cada cuadro de un video, usamos un modelo CNN de aprendizaje profundo para usar la extracción de características y proporcionar predicciones sobre los objetos que aparecen en el cuadro. Para cumplir con la detección de objetos en tiempo real, utilizamos el modelo YOLO. Este

es un modelo de CNN con la tarea de detectar y clasificar objetos en tiempo real, en comparación con otras redes artificiales YOLO ofrece un mayor rendimiento tanto en velocidad como en precisión.

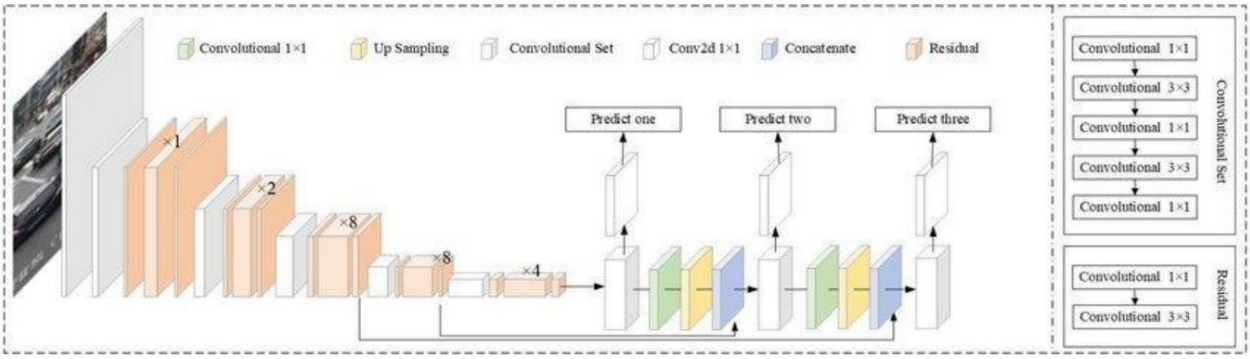

Figura 21. Modelo YOLOv3.

Entrada del modelo: girar los fotogramas del vídeo. Salida: contiene

cuadros delimitadores y objetos de cuadro delimitador respectivamente. Cada cuadro delimitador consta de 5 partes (x, y, w, h, predicción) siendo (x, y) las coordenadas del cuadro delimitador, (w, h) siendo el ancho y el alto del cuadro delimitador, respectivamente, predicción siendo la puntuación del objeto predicho.

Con el modelo que ha resuelto la primera parte, el problema es detectar los objetos en la imagen/video, que es la premisa para realizar el trazado y zonificación del objeto. Además, YOLO se divide en muchos modelos preentrenados para elegir. Desde el modelo más pesado (YOLOv5l) hasta modelos más pequeños como el YOLOv5n. Por supuesto, esto es una compensación, si desea una alta precisión, el tiempo de procesamiento debe ser largo. Este es también un tema que debe ser considerado experimentalmente.

Seguimiento de objetos Con la tarea de

rastrear objetos en cuadros de video, usamos el modelo Deep SORT[10] para ayudar a vincular objetos después de que hayan desaparecido por un tiempo.

Deep SORT es un modelo utilizado en la misión MOT. El modelo es realizado por nosotros en la siguiente secuencia:

- Utilice YOLO (descrito anteriormente) para detectar objetos en el marco actual
  en.
- Luego, use el filtro de Kalman para predecir nuevos estados de rastreo basados en rastreos anteriores.
  Estos estados en la inicialización asignarán valores exploratorios, si aún se garantiza que el valor
  permanecerá en los próximos 3 cuadros, el estado

transición del estado de sondeo al estado de confirmación e intentará mantener y monitorear los próximos 30 marcos. • Según los rastros confirmados, introdúzcalos en una cascada coincidente para asociar hallazgos, según la distancia y las métricas de características. • Manejo y depuración de los hallazgos y rastreo. • Utilice el filtro de Kalman para corregir el valor de la traza.

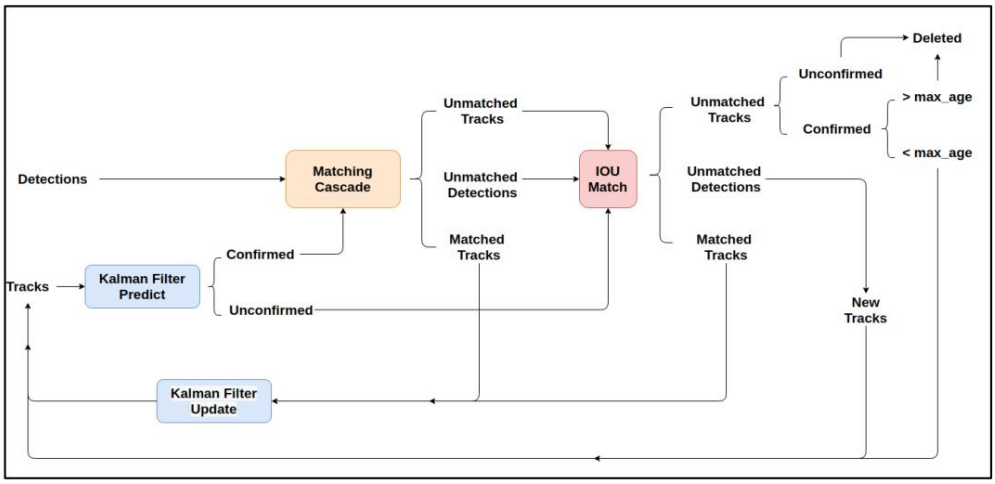

Figura 22. Flujo de procesamiento Deep SORT.

Además, el equipo eligió Deep SORT debido a su mayor precisión en tiempo real que los algoritmos de seguimiento. Aquí hay una tabla de comparación de algunos algoritmos de seguimiento:

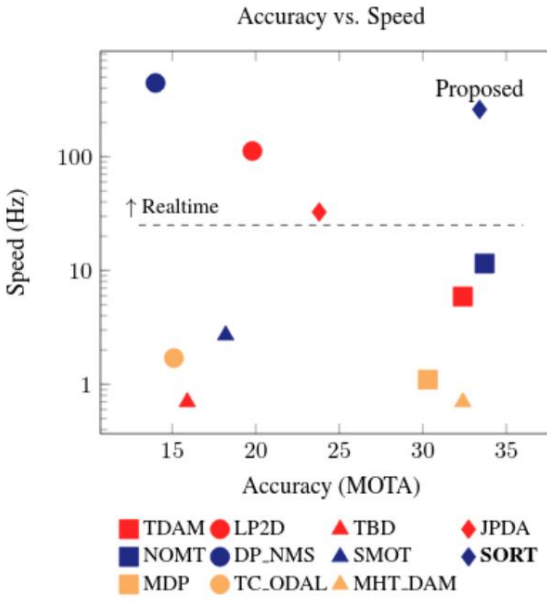

Figura 23. Comparación de algoritmos de seguimiento

6.2 Estimación de poses

Para realizar la clasificación de comportamientos extraños (o inusuales) en videos con precisión, usamos la estimación de forma antes de realizar el clasificador para considerar las formas que predecimos, ¿no son un comportamiento anormal o no?

Por ejemplo, si el cuadro anterior de una persona en un video realiza un comportamiento normal de caminar, gracias al seguimiento de este objeto, en el siguiente cuadro se detecta que este objeto está realizando un comportamiento de carrera, De repente, en este caso, puede ser consideró que en este cuadro aparece un comportamiento extraño y se hace una advertencia peligrosa.

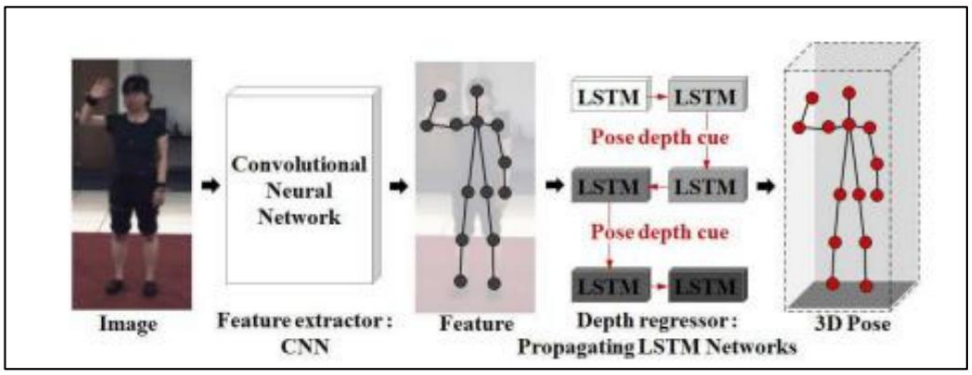

Figura 24. Procedimiento para usar LSTM para estimar la forma del cuerpo[11]

Al igual que otros módulos, Pose Estimation también debe tener características de procesamiento rápido y poder ejecutarse en tiempo real. Para lograrlo, el modelo MediaPipe desarrollado por el equipo de Google es la elección correcta para nosotros. MediaPipe se divide en dos fases que incluyen:

- Estime la forma 2D usando BlazePose[12]
- Luego estime la forma 3D usando GHUM[13]

BlazePose es un modelo para la estimación de la forma del cuerpo en 2D. La característica especial de BlazePose en comparación con otros modelos es la velocidad de procesamiento. Aquí, con la imagen de entrada, el modelo generará 33 puntos clave de ubicaciones en el cuerpo humano.

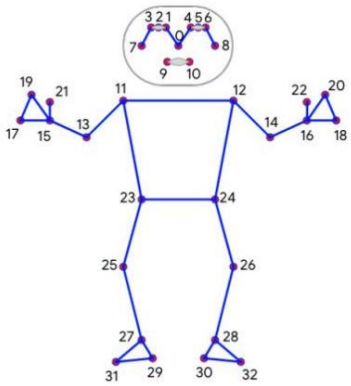

Figura 25. 33 puntos clave proporcionados por BlazePose

Para poder realizar una estimación de pose rápidamente, los autores han dividido este proceso en dos partes principales:

- Detección de rostros (detector de rostros)

• seguimiento de poses

Debido a que nuestro problema es estimar la forma del cuerpo para una secuencia de videos o imágenes directamente desde la cámara, la figura entre los dos cuadros no cambiará demasiado. En base a esa característica, en lugar de tener que volver a identificarse desde el principio, los autores solo necesitan volver a alinear el cuerpo después del primer reconocimiento.

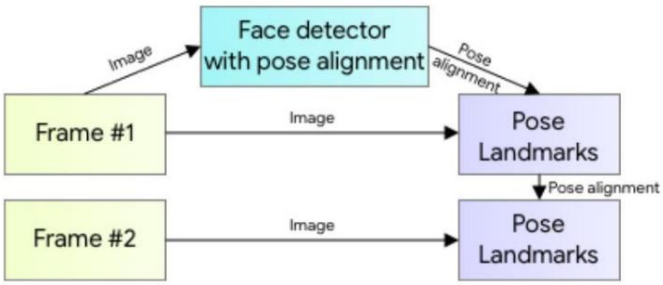

Figura 26. Proceso de estimación de pose de BlazePose.

En la actualidad, muchos modelos de detección de objetos utilizan la supresión no máxima (NMS) en el paso final para eliminar los cuadros delimitadores superpuestos. Sin embargo, esto también tiene muchas desventajas en diferentes contextos como darse la mano, abrazarse, etc. Por ello, los autores se centran en identificar una parte del cuerpo humano para estimar las posiciones restantes. Aquí, el equipo concluyó que los rostros humanos son la ubicación más reconocible para las redes neuronales, ya que tienen características de alto contraste y poca variación en la apariencia.

Para detectar rostros en tiempo real, el equipo de Google utiliza otro modelo de diseño propio, que es BlazeFace. Además, este modelo también proporciona algunos otros parámetros para la alineación, como el punto entre las caderas, el tamaño del círculo que rodea a la persona.

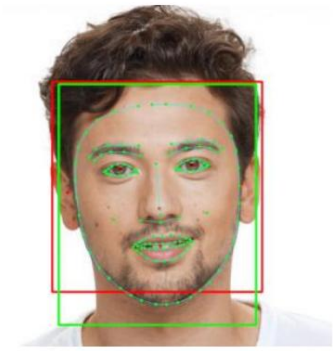

Figura 27. BlazeFace Detección de rostros

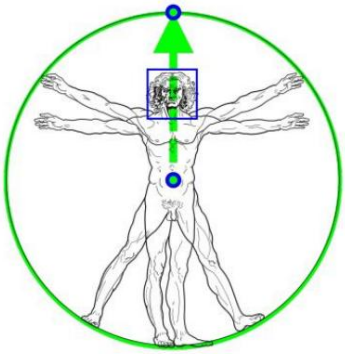

Figura 28. Alineación del hombre de Vitruman mediante reconocimiento facial

Finalmente, la arquitectura del modelo de BlazePose utiliza dos partes que incluyen mapas de calor, mapas de desplazamiento (izquierda); y una red de regresión (derecha). Durante el proceso de entrenamiento, los autores usaron primero los modelos izquierdo y medio. Luego, la nueva regresión se entrena compartiendo las características de la red de la izquierda, pero no la propagación hacia atrás para la red de la izquierda.

Durante la ejecución de la inferencia, la salida del modelo de mapa de calor se descartará y se usará solo la salida de la red de regresión.

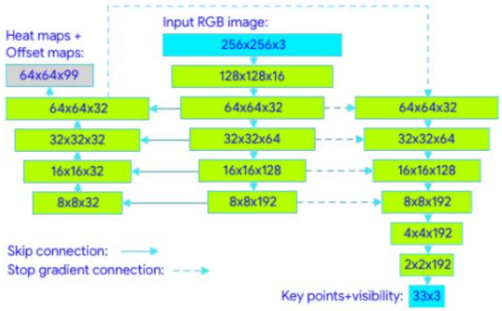

Figura 29. Modelo de BlazePose

Luego, el modelo GHUM se aplica en la salida de BlazePose. Este es un modelo muy complejo porque no solo usa información simple, sino que también usa modelos humanos escaneados en 3D para crear espacios de muestra.

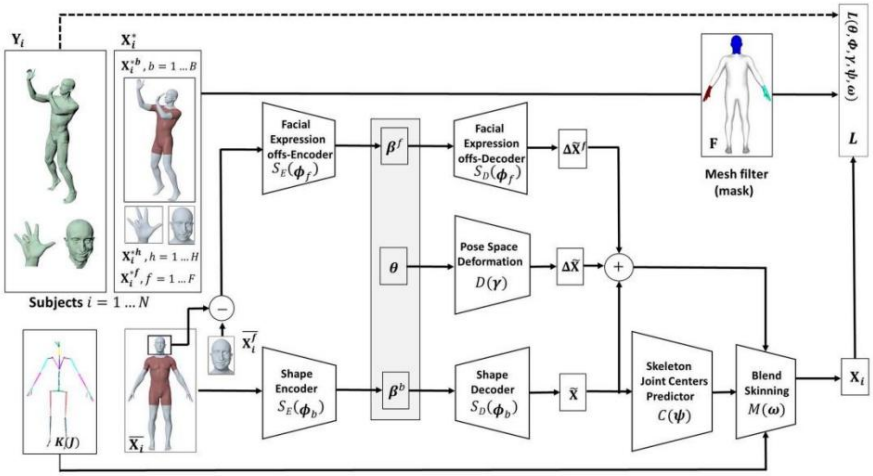

Figura 30. Modelo GHUM

6.3 Arquitectura modelo propuesta

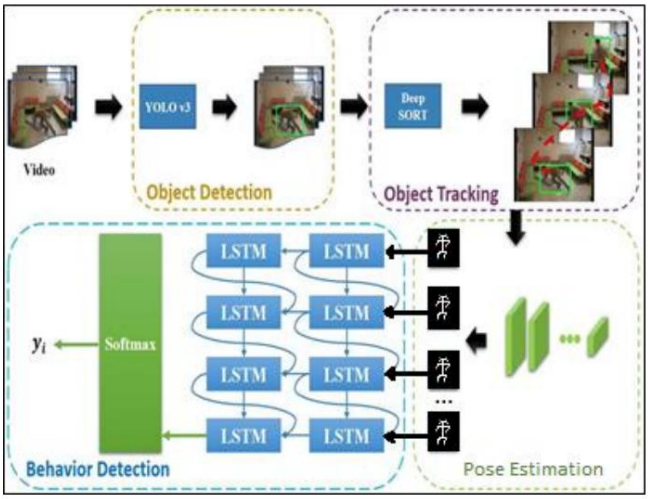

Figura 31. Modelo de detección de comportamientos extraños propuesto por el equipo

La arquitectura modelo propuesta incluye las siguientes fases:

- Detección y seguimiento de objetos.

• Para cada objeto detectado y rastreado, realice una estimación de forma. • Clasificar el comportamiento a partir de la forma estimada del cuerpo. • Zonificación de objetos con comportamiento extraño.

Para comprender mejor el modelo, partiremos del problema más básico de identificación para un solo objeto. Con este problema, podemos usar fácilmente la estimación de pose para reconocer la forma del cuerpo y usar LSTM para la clasificación. Sin embargo, con el problema más difícil: múltiples objetos, la estimación de pose será más difícil porque entonces no sabemos cuántos objetos hay. Para encargarse de esto, recomendamos usar la detección de objetos, específicamente YOLO. Después de conocer la posición de los sujetos, el siguiente problema es saber qué figura pertenece a quién en cada fotograma. Esto se resolverá asignando un identificador (ID) a cada persona en el marco de la imagen, este es el problema de seguimiento de objetos.

7 Experimentar y evaluar los resultados

7.1 Base de datos

Acerca del problema de seguimiento: podemos usar datos de un taller famoso en el campo de la visión artificial: ICCV 2021 MMPTRACK (iccv2021-mmp.github.io). Este conjunto de datos incluye videos de aproximadamente 5 horas para capacitación y 1,5 horas para validación. Todos los objetos en el video están marcados con cuadros delimitadores e identificadores predefinidos. Además, los videos se graban desde la cámara en diferentes ángulos. Las audiencias involucradas en el desarrollo de este video se distribuyeron en diferentes edades, géneros y razas.

Figura 32. Conjunto de datos MMPTRACK

Acerca del problema del reconocimiento de poses y la clasificación de comportamientos: MPII Human Pose es una base de datos de última generación para evaluar modelos del cuerpo humano y clasificar comportamientos. El conjunto de datos tiene más de 25.000 fotos y más de 40.000 personas con articulaciones corporales. Estas imágenes se recopilan todos los días con más de 410 comportamientos diferentes.

Podemos ir a la página de inicio de conjuntos de datos para ver algunos ejemplos. Las formas con borde rojo son las que se usan para entrenamiento, el resto se usará para prueba.

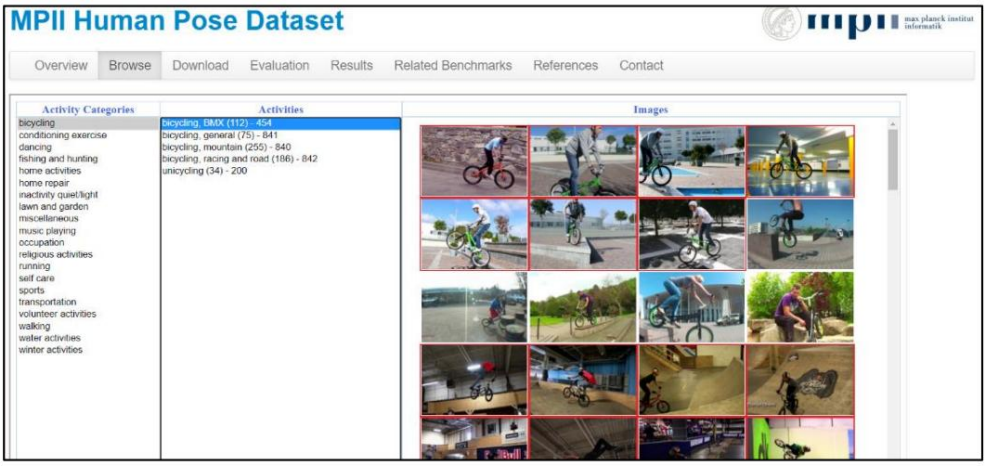

Figura 33. Conjunto de datos MPII Human Pose

7.2 Experimentación

Debido al límite de tiempo, el grupo de estudiantes construyó un pequeño experimento para el problema de clasificación de comportamiento por Estimación Pose. Estas son solo las dos últimas fases del modelo propuesto. Por lo tanto, el experimento solo puede operar en un sujeto. Cuando se ponen en práctica, las 4 fases ayudarán al modelo a trabajar en muchos objetos.

Para el problema de estimación de pose, el equipo eligió la biblioteca MediaPipe proporcionada por Google. Esta es una biblioteca instalada con los modelos BlazePose y GHUM mencionados anteriormente. La razón por la que el equipo eligió esta biblioteca es que tiene una velocidad de procesamiento muy rápida que se puede ejecutar en tiempo real.

Primero, diseñamos un programa que nos ayude a generar conjuntos de datos para entrenar y validar el modelo. En este programa, grabaremos 600 fotogramas continuos de imágenes para cada comportamiento y guardaremos los puntos de referencia de la estimación de pose en un archivo csv. En el siguiente ejemplo, estamos generando datos para la clase "Swing hand" en el cuadro 322.

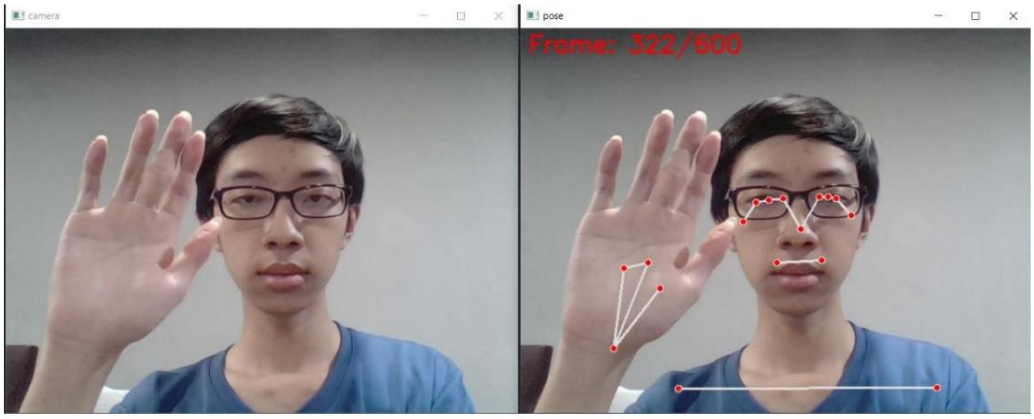

Figura 34. Proceso de Entrenamiento

Aquí, el grupo de prueba en 4 clases diferentes incluye:

- Normal (inactivo) •
- Ondulado (mango oscilante) •
- Cubierta ocular (cubierta ocular)

• Mango en W (W_hand)

Luego, el grupo entrena el modelo LSTM extrayendo 10 cuadros consecutivos como un punto de datos. El modelo LSTM de grupo experimental es un modelo simple que contiene 4 capas LSTM de 50 unidades. Además, para evitar el sobreajuste, se agregan capas de abandono después de cada capa LSTM. Finalmente, el modelo termina con una capa totalmente conectada cuyo número de unidades corresponde al número de clases. Obviamente, para que la salida sea una probabilidad distribuida, la función de activación sería softmax.

Layer (type)	Output Shape	Param #
lstm (LSTM)	(None, 10, 50)	36600
dropout (Dropout)	(None, 10, 50)	0
lstm_1 (LSTM)	(None, 10, 50)	20200
dropout_1 (Dropout)	(None, 10, 50)	0
lstm_2 (LSTM)	(None, 10, 50)	20200
dropout_2 (Dropout)	(None, 10, 50)	0
lstm_3 (LSTM)	(None, 50)	20200
dropout_3 (Dropout)	(None, 50)	0
dense (Dense)	(None, 4)	204

Figura 35. Módulo de estratificación LSTM.

El modelo anterior fue entrenado por el grupo con 32 épocas y dividió los datos en 64 lotes para optimizar utilizando el método de Adam. Y los resultados muestran que este modelo aprende muy bien, solo con las primeras 5 épocas, el modelo ha logrado una eficiencia muy alta.

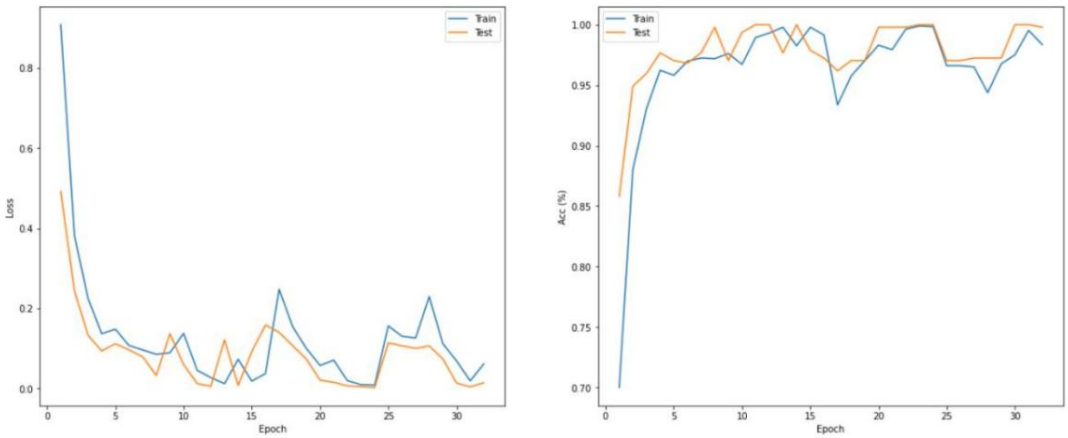

Figura 36. Resultados del entrenamiento

Finalmente, el equipo realiza una ejecución de inferencia en el modelo recién entrenado:

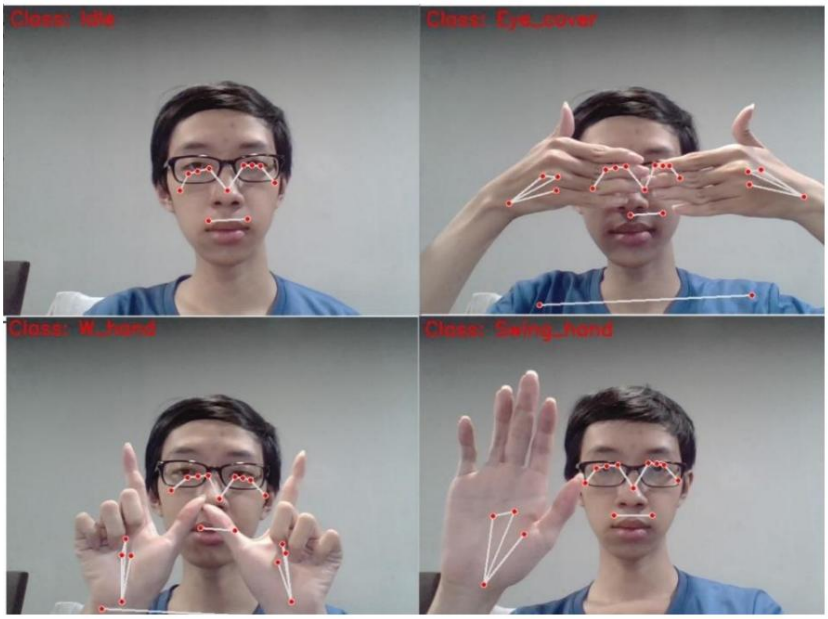

Figura 37. Ejecución de inferencia en tiempo real

7.3 Reseñas

En general, la velocidad y los resultados del modelo son bastante buenos. Con el modelo para un solo objeto, podemos ejecutar completamente en tiempo real. Al mismo tiempo, la aplicación del modelo YOLO no afectará el tiempo porque YOLO nació para ejecutarse en aplicaciones en tiempo real.

Por otro lado, hay algunas cuestiones pendientes que necesitan ser analizadas y resueltas para que el uso de la forma del cuerpo sea capaz de limitar algunos comportamientos. Por lo general, el comportamiento de levantar dos dedos y agitar se reconoce como el mismo porque los puntos de referencia no cambian. Para superar esta deficiencia, el equipo propondrá una solución más mejorada en la siguiente sección.

8 Conclusiones y direcciones para el desarrollo

8.1 Conclusiones Documento de

investigación sobre redes de aprendizaje profundo como CNN, RNN y LSTM para uso en el problema de detectar y zonificar objetos con comportamiento extraño en cámaras de CCTV.

El uso combinado de CNN y LSTM para realizar las tareas de detección, seguimiento y estimación de figuras humanas hace que la clasificación de comportamientos sea muy eficaz. Por lo tanto, se puede ver que es factible utilizar el modelo propuesto en el problema planteado.

8.2 Dirección de desarrollo

Los estudiantes continuarán investigando y desarrollando el modelo propuesto para mejorar la precisión y tiempo de ejecución, análisis de contenido.

Instalar en diferentes plataformas y realizar pruebas en el entorno real de las cámaras de CCTV para poder evaluar objetivamente el modelo propuesto.

Desarrolle el problema en muchos conjuntos de datos diferentes, en muchos entornos diferentes. Además de tipificar comportamientos inusuales en departamentos, edificios, etc., es posible diversificar el ambiente implementándolo en lugares donde se instalan cámaras para aumentar la seguridad, mejorar el rendimiento y reducir la fuerza de los equipos de monitoreo de seguridad en las áreas.

Además, en lugar de usar solo la forma del cuerpo usando Pose Estimation, el grupo de estudiantes también probará algunos modelos de CNN como MobileNet para extraer características de imágenes en tiempo real, mejorando así el modelo.

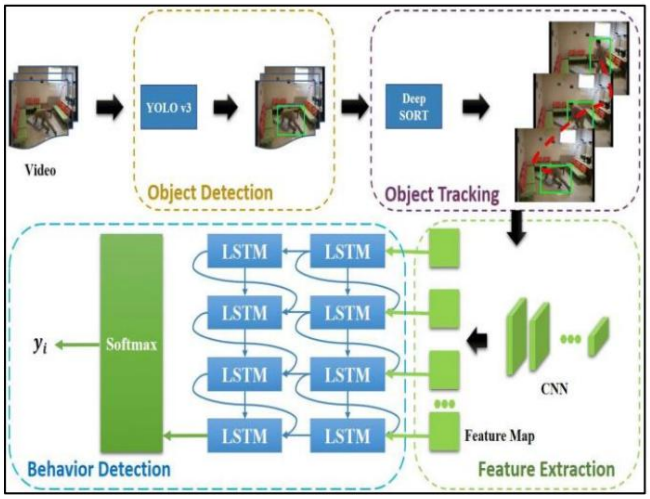

Figura 38. Modelo propuesto en dirección de desarrollo

9 Referencias

[1] Oluwatoyin P. Popoola y Kejun Wang. Comportamiento humano anormal basado en video Reconocimiento.

[2] Estimación de T. Wang, Q. Li, Y. Liu， Y.Zhou. Reconocimiento anormal del comportamiento del cuerpo humano usando pose

[3] Sandersan Onie et al., The Use of Closed-Circuit Television and Video in Suicide Prevention: Narrative Review and Future Directions, 7 de mayo de 2021 [4] Julak Lee et al., Application of sensor network system to prevent suicide from the puente, noviembre de 2016

[5] G. Spathoulas et al., Detección de comportamiento anormal en entornos de hogares inteligentes, IEEE, 21 de noviembre de 2019

[6] Cem Direkoglu et al., Detección de comportamiento anormal de multitudes mediante el uso de novelas basadas en flujo óptico características, IEEE, 23 de octubre de 2017

[7] Gioele Ciaparrone, Francisco Luque Sánchez, Siham Tabik, Luigi Troiano, Roberto Tagliaferri, Francisco Herrera. Aprendizaje profundo en el seguimiento de múltiples objetos en video: una encuesta [8] Damla Arifoglu, Abdelhamid Bouchachia. Reconocimiento de actividad y detección de comportamiento anormal con redes neuronales recurrentes.

[9] Chuan-Wang Chang, Chuan-Yu Chang y You-Ying Lin: un híbrido basado en CNN y LSTM modelo de aprendizaje profundo para la detección de comportamientos anormales.

[10] Nicolai Wojke, Alex Bewley, Dietrich Paulus. Deep SORT - Sencillo en línea y en tiempo real Seguimiento con una métrica de asociación profunda (2017)

[11] Kyoungoh Lee, Inwoong Lee y Sanghoon Lee. Propagación de LSTM: estimación de pose 3D basado en la Interdependencia Conjunta.

[12] Valentin Bazarevsky et al., BlazePose: seguimiento de la postura del cuerpo en tiempo real en el dispositivo, 17 de junio

2020

[13] Hongyi Xu et al., GHUM & GHUML: forma humana 3D generativa y pose articulada
Modelos, CVPR 2020