Passenger Counting in Mass Public Transport Systems using Computer Vision and Deep Learning

William David Moreno Rendón, Carolina Burgos Anillo, Daniel Jaramillo-Ramirez, *Member, IEEE*, , and Henry Carrillo

***Abstract*—Estimating the number of people in vehicles and stations in public transport systems is crucial to improve the quality of service. The TransMilenio system in Bogotá has serious drawbacks related to the lack of information in congestion situations. In this work, we present a computer vision method that estimates the number of people in TransMilenio's boarding platforms using deep learning techniques. We release the TransMilenio-Javeriana dataset with nearly 900,000 head labels on buses and stations. From these images, a deep learning architecture tuned for crowd counting was trained to generate density maps around the heads in the scene. Several head count methods were evaluated on the density maps. After testing these methods with 10,800 images, the results show a mean absolute error of 1 head per frame, equivalent to 11 % relative error. The accuracy found is much better than its manual counterpart. The proposed system is also scalable and low-cost, which indicates that it has great potential to provide information for the planning and operation of public transport systems.**

***Index Terms*—Automatic passenger counting, station boarding, public transport, convolutional neural networks, computer vision.**

I. Introducción

Bogotá es una ciudad capital con una población metropolitana cercana a 10 millones de habitantes y una muy alta densidad cercana a 24,000 habitantes por kilómetro cuadrado. No cuenta con una red férrea de transporte de pasajeros y su principal medio de transporte masivo es una red de buses de tránsito rápido (BRT, *Bus Rapid Transit*) conocida como TransMilenio (TM). El sistema TM, inaugurado en el año 2000, es reconocido mundialmente por desencadenar el boom del BRT moderno y ha sido replicado en cientos de ciudades. Este sistema considerado el BRT de mayor capacidad en el mundo, movía hasta antes de la pandemia cerca de 2.5 millones de pasajeros al día, alcanzando capacidades como las de sistemas férreos pesados (entre 20 y 50 mil pasajeros por hora por sentido), a una fracción de los costos de infraestructura y operación. Sin embargo, para lograr tan altas capacidades, TM requiere un uso intensivo de servicios expreso (rutas que no paran en todas las estaciones). Los 114 km de vías troncales (carriles exclusivos y estaciones con sobrepaso) son recorridos por 8 servicios corrientes (paran en todas las estaciones) de menor uso y unos 90 servicios expreso que usan múltiples plataformas de abordaje en cada una las 147 estaciones del sistema. En cada estación, todas las plataformas de abordaje y sus puertas son compartidas por 2, 3, o 4 rutas diferentes, generando imposibilidad de hacer fila, gran congestión de pasajeros, empujones en los abordajes y fuerte deterioro en la calidad de servicio. En horas pico, entrar en los servicios más congestionados puede llegar a ser físicamente imposible. Adicionalmente, el alto número de servicios expreso genera alta incertidumbre en el sistema, y dificulta su planeación y operación. Es más difícil estimar la ocupación de los buses, la congestión en las estaciones, los tiempos de espera y los transbordos internos, dificultando las acciones para mejorar la calidad del servicio.

William David Moreno Rendón, Carolina Burgos Anillo, Daniel Jaramillo-Ramirez and Henry Carrillo are with the Department of Electronics, Pontificia Universidad Javeriana e-mail: moreno_william, carolinaburgos, d-jaramillo and h.carrillo @javeriana.edu.co .

Visión por computador es una de las tecnologías que podría ayudar a reducir la falta de información y mejorar la calidad de servicio en las situaciones críticas de congestión en TM. Un sistema de visión por computador, con una mediana capacidad de cómputo en el borde, puede identificar cabezas, contar personas, estimar densidades, hacer seguimiento al movimiento y completar aforos de flujos tanto en buses como en estaciones. Diferentes técnicas de visión por computador han sido utilizadas en situaciones similares (ver sección A); no obstante, no se conocen trabajos en la literatura para estimar el número de personas en sistemas de transporte de alta capacidad, en escenas de alta densidad y en estaciones con plataformas de abordaje compartidas por varias rutas. Esta investigación también se aplica en situaciones similares que pueden ocurrir en muchos sistemas de transporte masivo, incluyendo sistemas férreos de alta capacidad, donde también es de gran importancia contar personas, estimar densidades o tiempos de espera.

A. Estado del Arte

Tradicionalmente se han utilizado conteos visuales realizados por humanos para estimar el número de pasajeros. Más recientemente también se han empleado métodos de conteo automático de pasajeros (APC - *Automatic Passenger Counting*), basados en varias tecnologías de adquisición de datos, entre ellas sensores de tapete, sensores infrarrojos, sensores de peso, distintas técnicas de visión por computador y sistemas que aprovechan el uso de dispositivos Wi-Fi [1]–[3].

El uso de sensores de carga posibilita el conteo de personas indirectamente. Gracias a los datos de peso ajustados según la acción de sistemas de frenos, se puede caracterizar estadísticamente la distribución de pasajeros, como se implementa desde 2013 en la red ferroviaria urbana de Copenhague. En esta ciudad se realizó una comparación de un método manual con el APC implementado y se obtuvo una alta correlación (0.9)

entre ambos. Los errores en la estimación son inherentes a la varianza del peso de cada persona y el uso de carga extra al peso promedio (maletas, bicicletas, etc) [4].

El conteo de personas mediante sensores infrarrojos se realiza frecuentemente en tiendas y edificios; con el fin de tener un control de flujo de entrada y salida de personas más eficiente. En sistemas de transporte público, su uso es limitado a sistemas de baja capacidad o sin congestión como en [2]. En cuanto a los métodos de conteo utilizando Wi-Fi, un buen ejemplo es el sistema iABACUS [1]. Este sistema reconoce las direcciones MAC de los dispositivos que se encuentren al alcance de un punto de acceso ubicado dentro del bus, sin necesidad de que los dispositivos se conecten. También permitiría seguir el dispositivo si se mueve dentro del sistema de transporte. La aleatorización de las direcciones MAC, tendencia generalizada en todos los fabricantes de teléfonos, dificulta fuertemente la efectividad de esta técnica.

En las últimas décadas, ha tomado fuerza también la técnica de contar pasajeros a partir de visión por computador y distintos métodos se han propuesto. En general, se pueden dividir en tres enfoques [5]. Un primer método es el conteo basado en *trajectory clustering*, este método se basa en la hipótesis de que las trayectorias que realiza un mismo cuerpo humano son más parecidas entre sí que con las de otros individuos, entonces se detectan características visuales y sus trayectorias son agrupadas en *clusters*, estimando el número de personas por el número de *clusters*. Un segundo método es el conteo basado en regresión, en este tipo se estima el número de personas por medio del aprendizaje de la función de regresión entre las características de las imágenes de entrada y las personas contadas en ella. Un tercer método se denomina conteo por detección, en este se selecciona un detector ya existente o se diseña uno para detectar a las personas en la imagen de entrada [3], [6]–[9]. En particular, el presente trabajo se enfoca en utilizar una técnica de detección de cabezas para hacer conteo de personas, en situaciones en las que podrían presentarse multitudes. Por lo tanto, se desarrolla una aplicación de *Deep Learning* para *Crowd Counting* en sistemas de transporte masivo. Los trabajos previamente mencionados, se enfocan principalmente en el problema del flujo de pasajeros, o el conteo de pasajeros en movimiento, dentro de buses o trenes, principalmente en el ingreso o el egreso del vehículo. Algunos de estos trabajos se enfocan en la adecuación de algoritmos y redes neuronales para la implementación en tiempo real [10]. Otros incluso son entrenados con imágenes de multitudes que no corresponden a escenas de transporte masivo, o con imágenes en ángulos susceptibles a oclusión [11], [12]. Este trabajo se diferencia de la literatura disponible principalmente por tres razones:

- No se hace seguimiento o *tracking* de pasajeros, por lo tanto, no se cuentan abordajes ni descensos. Solo se hace detección y conteo de cabezas en una imagen quieta, *frame por frame*.
- Las imágenes utilizadas no son al interior de vehículos, donde la altura de la cámara y las condiciones de espacio generan imágenes con pocas cabezas para contar y los flujos son de un pasajero a la vez. Las imágenes de este trabajo son tomadas en posición cenital, desde el techo de las plataformas de abordaje, logrando capturar en la imagen hasta 30 cabezas al tiempo.
- Este trabajo incluye la publicación de repositorio de videos con más de 200 horas de grabaciones, con 36,000 frames y cerca de 900,000 cabezas anotadas, que deben servir para el desarrollo de sistemas de seguimiento y conteo en cualquier sistema de transporte público de buses o trenes.

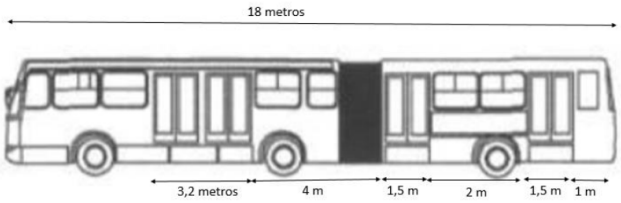

Fig. 1. Dimensiones de un bus articulado, tomado de [13].

Este trabajo sigue una metodología típica de los agentes de aprendizaje supervisado, adaptada a las necesidades de un sistema de visión por computador para el conteo de cabezas. Por lo tanto incluye las etapas de adquisición (repositorio de datos), anotación y pre-procesamiento, entrenamiento, detección y conteo de cabezas, y pruebas. La organización de este artículo se explica a continuación. En la sección II se describen los detalles de la implementación de un sistema de visión por computador en una estación de TM, su capacidad y sus restricciones técnicas. Además, se describe también el repositorio de videos TM-PUJ que contiene las grabaciones obtenidas. Este repositorio queda abierta a disposición de la comunidad de investigación. En la sección III se describe el pre-procesamiento realizado a las imágenes antes de ingresar la información a la red neuronal. En la sección IV se explica la red neuronal utilizada, así como el proceso de entrenamiento y validación. En la sección V se exponen los métodos de detección y conteo utilizados. En la sección VI se explican los resultados obtenidos para finalmente sacar conclusiones en la sección VII.

II. Descripción del repositorio de videos

A. Captura de la Información

En la configuración tradicional del sistema TM (hasta 2019), casi todas las estaciones incluían plataformas de abordaje para buses articulados (150 pasajeros), que conformaban la mayoría de la flota. Pocas estaciones podían recibir buses biarticulados (250 pasajeros). Los buses articulados (de 18 m de longitud) usan plataformas con 3 puertas: una puerta doble (3.2 m) en el vagón delantero y dos puertas sencillas (1.5 m cada una) en el vagón trasero, como se observa en la Figura 1.

Según la altura del techo de la estación y los ángulos de visión disponibles, se instaló en cada puerta sencilla una cámara, y en la puerta doble dos cámaras, completando 4 cámaras monoculares ELP-170, cuya distribución se observa en el plano de la Figura 2. Cada cámara va conectada a una mini-computadora Raspberry Pi B+ con una tarjeta SD de 64 GB, que según la luminosidad y el movimiento en la imagen, podría grabar video comprimido por aproximadamente 100

horas. Para evitar archivos muy pesados poco manipulables, cada 15 minutos o 27000 frames, la Raspberry cortaba la grabación y reiniciaba en un nuevo archivo (el video graba aproximadamente a 30 fps).

Fig. 2. Plano de la estación Polo año 2019 y ubicación de las cámaras, tomado del sitio web del sistema TransMilenio.

Los dispositivos realizaron grabaciones entre el miércoles 23 y martes 29 de mayo de 2019. Fueron instalados en la estación Polo de la troncal Calle 80, en la dirección occidente-oriente, sobre las primeras cuatro puertas orientales, que daban abordaje a los servicios H20, H17 y J24. Los clips obtenidos, con duración de 15 minutos, forman parte del repositorio de video denominado TransMilenio-Javeriana.

B. Estructuración del Repositorio

El repositorio TransMilenio-Javeriana [14] se compone de videos en buses y estaciones. Contiene 120 horas de video en un bus con diferentes niveles de iluminación y oclusión incluyendo 4 perspectivas diferentes: una cámara ubicada a la izquierda de una puerta doble del bus, otra ubicada a la derecha, otra en el centro de la puerta intermedia y finalmente una cámara en el centro de la puerta trasera. En total se obtuvieron 480 video-clips en el bus. Además, el repositorio contiene 107 horas de video en estaciones con diferentes niveles de iluminación y oclusión, comprendidas en 3 perspectivas de la puerta: una donde la cámara está a la izquierda de la puerta doble, otra con la cámara a la derecha y finalmente la cámara que filma una puerta sencilla desde el centro. Todas las cámaras tienen vista superior (cenital). En la estación se obtuvieron 430 videos. Sumando la estación y el bus, se registraron 227 horas de grabación, en 910 videos con una duración promedio de 15 minutos cada uno.

Para esta investigación se utilizaron 20 video-clips en la estación, cada uno tiene 1800 frames. El total de las imágenes estudiadas, así como los detalles del repositorio, se resumen en la Tabla I.

Para facilitar el análisis, mejorar la efectividad de los métodos de conteo y revisar en qué tipo de escenas el algoritmo es más efectivo, se establecen 4 niveles de ocupación. La descripción cualitativa de cada nivel se presenta a continuación:

- **Nivel 1:** Ocupación baja. Estación casi vacía. Imágenes que registran un máximo de 3 personas.

TABLA I
RESUMEN DEL REPOSITORIO DE VIDEO TRANSMILENIO-JAVERIANA Y LOS ARCHIVOS UTILIZADOS EN ESTE ESTUDIO.

Repositorio completo		
Lugar	Horas de grabación	# Video-clips (15 min)
Bus	120	480
Estación	107	430

Estudio presentado (frames usados)

Videoclips (1 min)	# Cabezas anotadas	# Frames anotados	Entrenamiento	Validación	Prueba
20	895231	36000	22660 (63 %)	2520 (7 %)	10800 (30 %)

- **Nivel 2:** Ocupación media-baja. Estación con poca afluencia de personas. Imágenes desde 4 hasta 9 personas.
- **Nivel 3:** Ocupación media-alta. Estación con imágenes de entre 10 y 25 personas.
- **Nivel 4:** Ocupación alta. Estación con obstrucción completa del paso en las puertas y dificultad para moverse en la apertura y cierre de puertas. Más de 25 personas por imagen.

La distribución de los frames de prueba según su nivel de ocupación se muestra en la Fig. 3, donde se puede apreciar un balance relativo de todos los niveles.

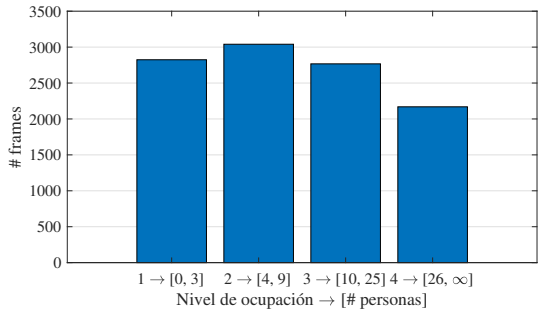

Fig. 3. Histograma de número de frames vs nivel de ocupación, para los 10800 frames de prueba.

III. PRE-PROCESAMIENTO

En esta etapa, las imágenes adquiridas deben ser inicialmente anotadas y posteriormente procesadas para poder entrar en la red neuronal. Para completar las anotaciones de los 20 video-clips definidos se usó el programa CVAT, que ayuda al usuario en la anotación de frames consecutivos, sugiriendo la ubicación de las etiquetas. De esta forma se completaron 36000 imágenes individuales anotadas, que en total contienen 895231 cabezas.

A. Generación de Datos para Entrenamiento

El conteo de multitudes (*crowd counting*) busca contar el número de personas en una escena concurrida. Uno de los métodos para realizar esta tarea es la estimación de densidad, que tiene como objetivo convertir una imagen concurrida en su

correspondiente mapa de densidad, que sirve para estimar el número de personas en la imagen. En [15] se presenta una red neuronal que estima un mapa de densidad de la ubicación de las cabezas en una multitud. Este modelo aprende de una serie de *kernels*, como se observa en la Fig. 4, donde identifican la ubicación de las cabezas de las personas en diferentes escalas. En el mapa de densidad, cada cabeza es convertida en una distribución normal en dos dimensiones.

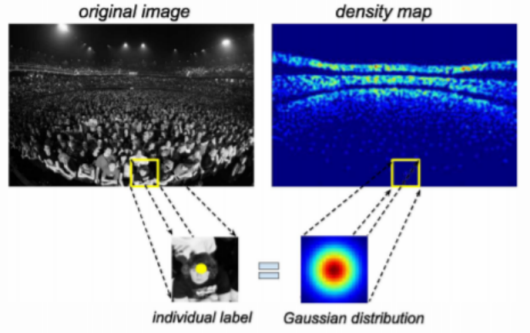

Fig. 4. Uso de kernel en imagen original para obtener una distribución normal en el mapa de densidad presentado en [15].

En este trabajo se utilizó un kernel Gaussiano en dos dimensiones descrito en la ecuación (1). Las variables x y y determinan la posición del kernel, mientras que σ es la desviación estándar de la distribución Gaussiana, que aproxima el ancho del kernel. Al aplicar el kernel Gaussiano, la imagen será más borrosa mientras más grande sea el valor de σ.

$$G_\sigma(x, y; \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{1}$$

Para usar kernels Gaussianos en dos dimensiones, es necesario mantener un compromiso entre la distorsión de la imagen filtrada y la precisión con la que el kernel indica una cabeza. Por lo tanto se analizaron dos alternativas: kernel fijo (con σ fijo) y kernel variable.

A1. Kernel fijo: En primer lugar, se realizó una adaptación del método de generación de mapas utilizado en la implementación original de [16]. Se usó un tamaño de ventana de 29x29 píxeles y $\sigma = 8$. El resultado se muestra en la anotación de la Figura 5 con una imagen de una estación de nuestro repositorio.

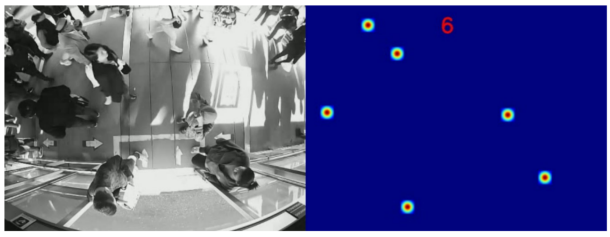

Fig. 5. Mapa de densidad con kernel fijo.

A2. Kernel variable: Esta opción consiste en usar un kernel Gaussiano sobre el punto central de las anotaciones pero esta vez, cambiar el tamaño y el parámetro σ de cada kernel considerando sus vecinos más cercanos. Para realizar esta función se partió del filtro Gaussiano implementado en [17], usando las distancias a los vecinos más cercanos para calcular el valor de σ para cada kernel. El resultado con las anotaciones realizadas se observa en la Figura 6.

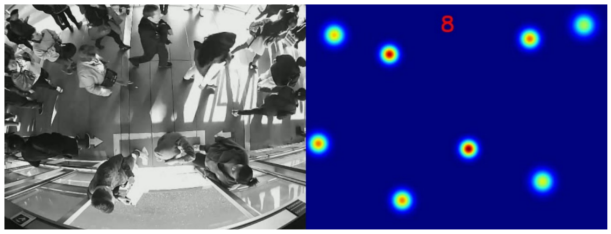

Fig. 6. Mapa de densidad con kernel variable.

Adicionalmente, se definió una región de interés (ROI) sobre las imágenes para eliminar los puntos que se encuentran cerca de los bordes donde las cabezas son muy pequeñas y en ocasiones indistinguibles, considerando que pueden generar ruido en los datos. En este caso la ROI está definida entre los píxeles 25 y 615 en x y entre los píxeles 40 y 465 en y, teniendo en cuenta que las imágenes originales tienen un tamaño de 640x480. El límite de la ROI se observa con una línea roja punteada en la Figura 9.

IV. Entrenamiento

Tras evaluar varias configuraciones de redes neuronales y sus resultados en aplicaciones de conteos de cabezas y conteos de multitudes (Yahiaoui 2010 [8], Lumentut 2015 [7], Zhang 2016 [18], Perng 2016 [6], Li 2016 [9], Sun 2019 [5], Labit-bonis 2021 [10], Kim 2022 [11],ver Tabla III), se eligió la implementación realizada en [18]. Es una red neuronal convolucional multicolumna (MCNN) de 5 capas, que implementa filtros en cada columna para adaptar los mapas de densidad correspondientes a cabezas en diferentes escalas.

Teniendo en cuenta que una red de aprendizaje profundo debe ser entrenada de manera iterativa, es necesario tener una función de pérdida. La función de pérdida por excelencia utilizada en los problemas de regresión es el error cuadrático medio (MSE), que se puede observar en la ecuación (2). También puede usarse el error absoluto promedio (MAE), como se muestra en la ecuación (3).

$$\text{MSE} = \frac{\sum_{i=0}^{n} \left(Y_i - \hat{Y}_i\right)^2}{n} \tag{2}$$

$$\text{MAE} = \frac{\sum_{i=0}^{n} \left|Y_i - \hat{Y}_i\right|}{n} \tag{3}$$

Durante el entrenamiento se completó también el proceso de ajuste o *fine tuning* cambiando diferentes hiper-parámetros en el modelo. Usando el MAE se encontró que los mapas de densidad con kernel variable no reducen la función de pérdida. El mejor resultado de entrenamiento se obtuvo con los mapas densidad de kernel fijo. Para estimar el número de cabezas en la imagen, se hace la suma de todos los píxeles del mapa de densidad.

A continuación se presenta un resumen de los parámetros de entrenamiento y validación obtenidos de manera heurística:

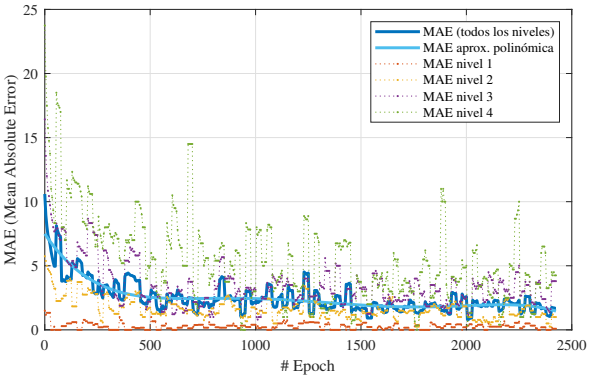

Fig. 7. Número de epochs vs MAE según el nivel de ocupación en la imagen.

- **Factor de escala para *Ground Truth* (GT):** Los mapas de densidad de entrada se multiplican por 1.1 para generar una mayor intensidad en el kernel y un menor MAE en el entrenamiento.
- **Batches:** Los 22660 frames de entrenamiento, son divididos en 63 lotes, evitando sobrecargar la memoria del computador.
- **Epochs per batch:** Se llevan a cabo 20 iteraciones sobre cada lote, evitando desbordar la capacidad de procesamiento.
- **Learning rate:** El tamaño del paso para cada iteración buscando encontrar un mínimo en la función de pérdida. Su valor fue 0.9, encontrado de forma heurística.
- **Epochs totales:** Número de iteraciones que se realizan del modelo durante el entrenamiento que en este caso es de 2450.

En la Figura 7 se observan las tendencias del MAE durante el entrenamiento, usando las imágenes de validación. Cabe resaltar que se utiliza el MAE como medida de desempeño del aprendizaje en función del número de *epochs*. Después de aproximadamente 2500 epochs, la red mostró signos de sobre-entrenamiento.

V. Detección y Conteo de Cabezas

Para obtener una mejor estimación en el número de personas representadas en los mapas de densidad de salida de la red, es necesario explorar diferentes métodos de conteo para las escenas obtenidas y los diferentes niveles de ocupación que en cada imagen.

A. Métodos de Conteo

A1. Conteo de la integral: La sumatoria de valores de los píxeles de la imagen en escala de grises dada por la ecuación (4), donde $f(x_i)$ es la imagen del mapa de densidad de n píxeles y x_i es el valor de la imagen en cada píxel.

$$\sum_{i=1}^{n} f(x_i) \tag{4}$$

A2. Conteo por picos: Mediante el método de binarización de Otsu se encuentra para cada imagen un umbral que determina aquellos píxeles que pueden representar picos en la imagen. Luego, se realiza una búsqueda del máximo pico local, por medio de una dilatación de la imagen como se indica en la ecuación 5, donde $f(x)$ es la imagen original y $f(x) \oplus Y$ la imagen dilatada. Seguido, los píxeles que cumplen esta condición en el radio de una cabeza, son agrupados en un solo pico; evitando que una sola cabeza contenga varios picos y sea contada más de una vez.

$$(f(x) \oplus Y) == (f(x)) \tag{5}$$

De forma preliminar, los métodos de conteo presentaron mejores resultados en ciertos niveles de ocupación pero no fueron efectivos en todos. Por lo tanto, se definieron dos nuevos métodos de conteo, combinando los métodos previos de la siguiente manera.

A3. Método combinado 1: Para conteos de menos de 7 cabezas se usa el método de la integral y para conteos de 8 o más cabezas se usa el conteo por picos. En la práctica ambos métodos se ejecutan en paralelo y se escogen dependiendo del valor determinado por ambos. En caso de contradicción se escoge el método de picos.

A4. Método combinado 2: Se utilizan los mismos rangos que en el método anterior, pero para conteos entre 4 y 17 cabezas, se cuentan los píxeles con un valor diferente a cero (píxeles que representan posibles cabezas) y se divide por el número de píxeles estimado en cada cabeza.

B. Evaluación de Métodos de Conteo

Para evaluar con mejor proporcionalidad los métodos de conteo, se utilizó el error relativo promedio porcentual, conocido como MRE:

$$\text{MRE} = \frac{100}{n} \sum_{i=0}^{n} \frac{\left|Y_i - \hat{Y}_i\right|}{Y_i}, \tag{6}$$

donde Y_i es el valor de referencia, $\hat{Y}_i$ es el resultado del conteo y n el número de frames considerados. Cuando el valor de referencia (GT) es $Y_i = 0$ (cerca del 10 % de los frames), para evitar la división por 0, el MRE se reemplaza por el valor promedio del MRE en aquellos casos con el mismo MAE $|Y_i - \hat{Y}_i|$ pero con $Y_i \neq 0$.

De esta forma se puede evaluar el desempeño en dos dimensiones: el error absoluto (MAE) y el error relativo (MRE) y determinar el mejor método como aquel que entregue la menor combinación de ambos errores para todos los niveles de ocupación de la estación.

En la Fig. 8, se observan gráficos de dispersión para el MAE y el MRE de la estimación del número de cabezas a partir de los 4 métodos descritos anteriormente. Cada uno de los puntos corresponde a un promedio del error sobre 108 imágenes diferentes, obteniendo en total 100 puntos para cada método, cubriendo así la totalidad de las 10800 imágenes de prueba para las estaciones. Es importante notar primero las diferentes escalas de las cuatro gráficas, tanto en el MAE,

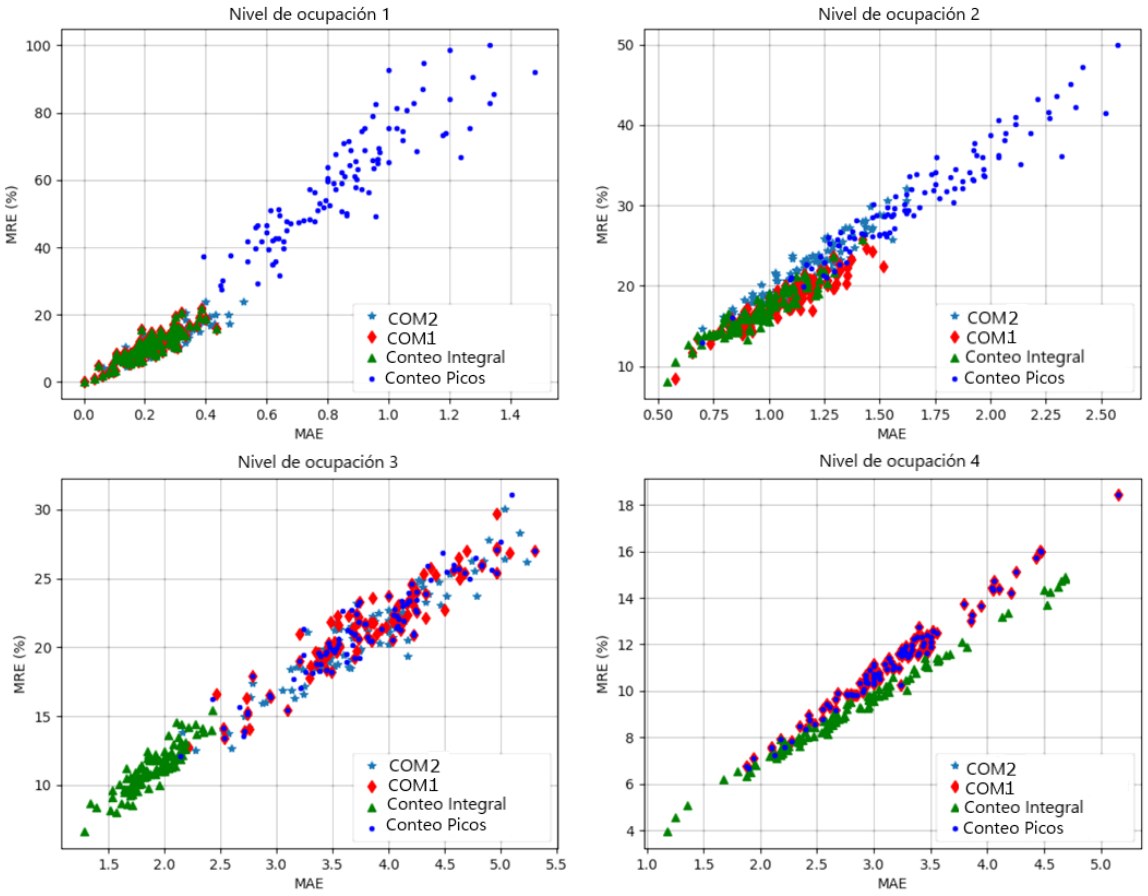

Fig. 8. Comparación MAE y MRE (%) para cada nivel de ocupación, con todos los métodos de conteo. Cada punto representa el promedio de 108 frames con el mismo nivel de ocupación. En total hay 100 puntos en cada imagen incluyendo todos los frames de prueba.

como en el MRE. Como es natural, el error absoluto aumenta a mayor nivel de ocupación, mientras el error relativo disminuye. El mejor resultado para el nivel 1 lo logra el método de conteo de la integral con un MAE de 0.22 y un MRE de 10.59 %. En el nivel 2 también el conteo de la integral es el mejor con un MAE de 0.96, un MRE de 16.9 %. En los primeros dos niveles, el conteo de picos tiene alto número de errores absolutos y relativos y los conteos combinados están más cerca del conteo de la integral. En el nivel 3 el conteo de la integral es muy superior a los demás con un MAE de 1.90 y un MRE de 11.23 %, mientras el conteo de picos arrastra los métodos combinados hacia 4 errores por frame y 25 % de MRE. Finalmente, en las imágenes del último nivel el resultado se observa el gráfico inferior derecho y muestra que el conteo de la integral es ligeramente mejor que los otros métodos con un MAE de 2.83 y un MRE de 9.29 %, siendo este el mejor método para los 4 niveles presentados en imágenes en estación. Los métodos combinados entregaron buenos resultados en redes entrenadas con imágenes tanto de estaciones y buses, pero en este trabajo, donde la red fue entrenada solo con imágenes en estación, los métodos combinados no superan al método de conteo de picos en ninguno de los niveles de ocupación.

Teniendo en cuenta los resultados proporcionados por los diferentes métodos de conteos básicos y combinados, el mejor resultado se obtiene con el método de conteo de la integral como se muestra por niveles de ocupación en la Tabla II. Promediando los resultados de todos los niveles, el método de la integral presenta un MAE de 1.39, un MRE de 14.24 % y un MSE de 2.07, indicando esto que el error en el conteo sobre los mapas generados está casi siempre y sin importar el nivel de ocupación entre 1 y 2 cabezas.

TABLA II
RESULTADOS MAE Y MRE PARA CONTEO DE LA INTEGRAL POR NIVELES DE OCUPACIÓN EN FRAMES DE PRUEBA.

Nivel de ocupación	Rango de cabezas	MAE	MRE (%)
1	0 - 3	0.22	10.6
2	4 - 9	0.96	16.9
3	10 - 25	1.90	11.2
4	$>$ 25	2.83	9.3

Para tener mejor comprensión del funcionamiento del sistema en términos visuales, se muestran ejemplos en la Figura 9. Para las imágenes previamente anotadas, se presenta a la izquierda la imagen original junto con el número de cabezas presentes en las anotaciones (AN), en el centro el mapa de densidad que se toma como *ground truth* (GT) con el conteo proporcionado al hacer la integral sobre este, y a la derecha se

presenta el mapa de densidad generado por la red superpuesto con la imagen original para poder visualizar dónde ubica el mapa las cabezas, acompañado del estimado obtenido con el método de conteo (ET). En el repositorio anexo de este artículo se encuentra disponible también un video con algunos resultados de ejemplo para su visualización.

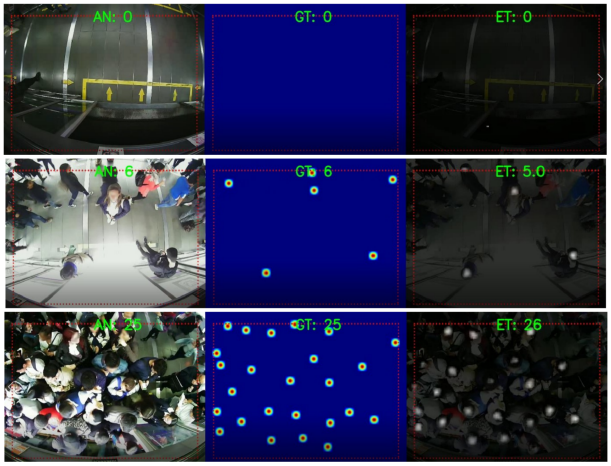

Fig. 9. Evaluación del modelo con estación vacía, media y llena según valor de GT (conteo estimado de *ground truth* o etiquetado) y ET (valor estimado obtenido con el método de conteo de la integral).

VI. Resultados y análisis

Para los clips de videos de prueba que no se encuentran anotados, se presenta a la izquierda la imagen original, en el centro el mapa entregado por la red superpuesto con la imagen y a la derecha el mapa visualizado como mapa de calor junto con el valor estimado. Dado que no se tienen anotaciones de estas imágenes, se escogieron escenas donde es fácil contar visualmente las personas presentes. En la Fig. 10 se pueden ver imágenes individuales que hacen parte de algunos de estos videos.

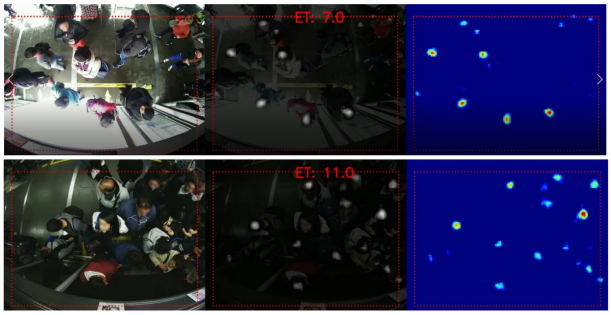

Fig. 10. Ejemplo de conteo con imágenes sin anotar (sin GT) en estación con poca y media congestión.

Tras completar el entrenamiento, la validación y encontrar el mejor método de conteo, se presentan los resultados obtenidos con las imágenes de prueba: 10800 frames seleccionados aleatoriamente de los 20 clips. Los resultados muestran una dinámica temporal inherente. Al organizar los frames de forma secuencial, dependiendo de los niveles de oclusión e iluminación, algunas partes de la secuencia pueden ser más difíciles para la detección y el conteo, generando sub-conteo (falsos negativos) o sobre-conteo (falsos positivos). Si son simultáneos, ambos errores se compensan. En la Figura 11 se observa cómo a pesar de que el conteo es independiente en cada frame y no tiene ningún método de seguimiento en la secuencia, el resultado es muy cercano al GT en todos los niveles de ocupación. Incluso los buenos resultados se mantienen cuando de manera abrupta un gran número de personas sale de la imagen (tras abordar un bus), como se ve en el recuadro Çlip 10, Nivel 2-3". A pesar de los buenos resultados, en esa misma secuencia el conteo no registra cerca de 3 de las cabezas en la escena, generando un MRE alto para ese video-clip.

Luego de analizar todos los resultados de imágenes de prueba en los 20 video-clips, solo uno de ellos presentó un resultado atípico y fue por tanto excluido del análisis. Adicionalmente, al revisar en detalle la Figura 11, en los cuatro gráficos superiores se nota cómo la señal de conteo muestra oscilaciones rápidas alrededor de un valor de GT que muchas veces es constante durante muchos frames. Para reducir estos errores se utilizó un filtro de moda móvil de 3 posiciones, que ayuda a disminuir el error total en algunas décimas porcentuales en todos los video-clips. También en la parte inferior de la Figura 11, se observa para cada nivel de ocupación el histograma del error absoluto en cada frame. El 39 % del total de los frames tienen un conteo perfecto (el error absoluto es cero). Adicionalmente se confirma la necesidad de reducir la varianza del error, probablemente a través de un mayor entrenamiento de la red, especialmente en los clips cuyo nivel predominante es el nivel 3.

Finalmente, los resultados de los frames de prueba para los 19 video-clips se observan en la gráfica de dispersión de la Figura 12. Se denomina nivel predominante en un clip, al nivel de ocupación obtenido tras redondear la media del nivel de ocupación de cada frame del clip. Los clips con nivel predominante 3 muestran los valores más altos de error absoluto, siempre por debajo de 3. Todos los valores de error relativo están por debajo del 25 % y solo dos de los clips superan el 16 %. En los resultados promedio sobre el total de los frames de prueba, muestran cómo el error absoluto está al rededor de 1 cabeza por frame, mientras el error relativo está cerca al 11 %. Esto implica que el sistema propuesto logra distinguir los niveles de ocupación con muy alta efectividad y el error encontrado es marginal para la planeación y operación de un sistema de transporte público en condiciones similares de alta capacidad.

A. *Comparación de Resultados*

Finalmente, para poner nuestro trabajo en contexto, se comparan los resultados obtenidos con otros de trabajos previos que se enfocan en situaciones similares. La Tabla III resume los trabajos similares, con las diferentes métricas consideradas por los respectivos autores: Exactitud (*Accuracy*), Precisión y Exhaustividad (*Precision and Recall*) y el Error Absoluto Medio (*MAE*). El resultado de este trabajo se muestra en la última línea. La exactitud (Accuracy) de 89 % y MAE de 1.01 obtenido con 20 video-clips del repositorio TransMilenio-

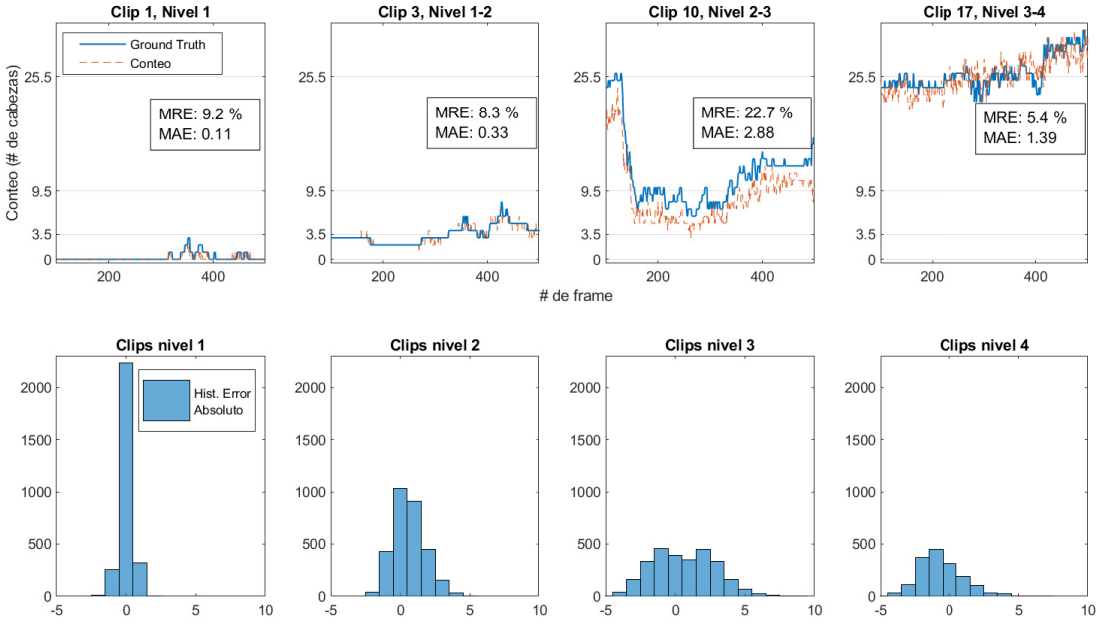

Fig. 11. Arriba: Ejemplos de conteos en video-clips con todos los niveles de ocupación con sus respectivos MAE y MRE. Abajo: histogramas del error absoluto para todos los frames según su nivel de ocupación.

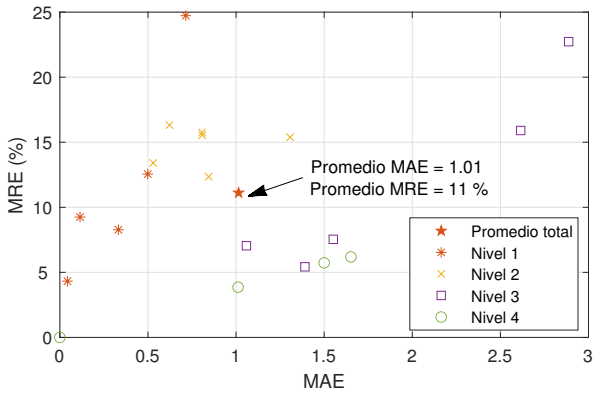

Fig. 12. MAE vs MRE para los frames de prueba en los 19 video-clips analizados, según su nivel de ocupación dominante.

Javeriana adaptando mapas de densidad y la red convolucional MCNN.

VII. Conclusiones

Aunque es evidente la mala calidad de servicio ofrecida en medios de transporte público masivo, es difícil enfocar esfuerzos en la planeación u operación del sistema por la falta de información precisa sobre la congestión de pasajeros en vehículos y estaciones. Ante la dificultad de hacer conteos de personas por métodos tradicionales, este trabajo presenta un método de visión por computador, efectivo, fácil de implementar, escalable y de bajo costo.

A partir de una base de datos de más de 100 horas de grabación, y casi 900,000 cabezas etiquetadas en 36,000 imágenes (frames), se entrenó y se evaluó el desempeño de una red neuronal de aprendizaje profundo para estimar el número de personas usando mapas de densidad con kernels Gaussianos.

Después de revisar cuatro métodos diferentes de conteo, el conteo de la integral muestra un error promedio de 1 cabeza por frame, correspondiente a un error relativo del 11 % equivalente a una precisión del 89 %. Esto fue obtenido en un conjunto de 10800 frames de prueba con cuatro niveles de ocupación diferentes.

El método probado tiene una capacidad de conteo mucho mejor que métodos tradicionales basados en aforos manuales, y funciona en diferentes condiciones de iluminación, oclusión y congestión. La precisión encontrada de 89 %, con un error medio de 1 cabeza por frame, permite estimar muy bien rangos de ocupación y densidades en las áreas del campo de visión de las cámaras instaladas, aportando valiosa información tanto para la planeación del sistema a largo plazo, como para la operación del mismo en el corto plazo.

Los resultados obtenidos muestran una relación natural del número de cabezas en frames sucesivos. Esta información que no ha sido aprovechada en este trabajo, podría ayudar a reducir el error promedio encontrado y permitiría además estimar correctamente los flujos de personas en movimiento, para calcular por ejemplo, el perfil de ocupación de los vehículos en tiempo real.

TABLA III
RESULTADOS COMPARATIVOS DE DIFERENTES MÉTODOS PARA CONTEO DE CABEZAS PRINCIPALMENTE EN TRANSPORTE PÚBLICO.

Publicación	Situación	Método	Resultado
Yahiaoui 2010 [8]	Flujo en bus,	Stereovision. Segmentation, binarization, tracking	Accuracy 97 %
Lumentut 2015 [7]	Flujo en estación	Counting (AMF)	Recall 88 %, Precision 19 %
Zhang 2016 [18]	Conteo de multitudes	Counting (CNN)	MAE 1.60
Perng 2016 [6]	Flujo en bus	Background subst., detection, tracking	Accuracy 87 %
Li 2016 [9]	Flujo en bus	RGB+D. Detection and tracking	Accuracy 92 %
Sun 2019 [5]	Flujo en bus	RGB+D. 3D body projection	Precision and Recall (92 %)
Labit-bonis 2021 [10]	Flujo en bus	YOLOv5 + DeepSORT	Accuracy 95 %
Kim 2022 [11]	Flujo en bus	Real-time YOLOv3	Accuracy 99 %
Moreno 2022	Conteo en estación	Density maps, MCNN	Accuracy 89 % MAE 1.01

REFERENCIAS

[1] M. Nitti, F. Pinna, L. Pintor, V. Pilloni, and B. Barabino, "Iabacus: A Wi-Fi-based automatic bus passenger counting system," *Energies*, vol. 13, no. 6, 2020.

[2] A. Olivo, G. Maternini, and B. Barabino, "Empirical study on the accuracy and precision of automatic passenger counting in european bus services," *The Open Transportation Journal*, vol. 13, no. 1, 2019.

[3] I. Grgurević, K. Juršić, and V. Rajič, "Review of automatic passenger counting systems in public urban transport," in *5th EAI International Conference on Management of Manufacturing Systems*, pp. 1–15, Springer, 2022.

[4] B. F. Nielsen, L. Frølich, O. A. Nielsen, and D. Filges, "Estimating passenger numbers in trains using existing weighing capabilities," *Transportmetrica A: Transport Science*, vol. 10, pp. 502–517, jul 2014.

[5] S. Sun, N. Akhtar, H. Song, C. Zhang, J. Li, and A. Mian, "Benchmark Data and Method for Real-Time People Counting in Cluttered Scenes Using Depth Sensors," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, pp. 3599–3612, oct 2019.

[6] J.-W. Perng, T.-Y. Wang, Y.-W. Hsu, and B.-F. Wu, "The design and implementation of a vision-based people counting system in buses," in *2016 International conference on system science and engineering (ICSSE)*, pp. 1–3, IEEE, 2016.

[7] J. S. Lumentut, F. E. Gunawan, *et al.*, "Evaluation of recursive background subtraction algorithms for real-time passenger counting at bus rapid transit system," *Procedia Computer Science*, vol. 59, pp. 445–453, 2015.

[8] T. Yahiaoui, L. Khoudour, and C. Meurie, "Real-time passenger counting in buses using dense stereovision," *Journal of Electronic Imaging*, vol. 19, no. 3, p. 031202, 2010.

[9] F. Li, F. Yang, H. Liang, and W. Yang, "Automatic passenger counting system for bus based on rgb-d video," in *2nd Annual International Conference on Electronics, Electrical Engineering and Information Science, EEEIS*, 2016.

[10] C. Labit-Bonis, J. Thomas, and F. Lerasle, "Visual and automatic bus passenger counting based on a deep tracking-by-detection system." working paper or preprint, Oct. 2021.

[11] H. Kim, M.-K. Sohn, and S.-H. Lee, "Development of a real-time automatic passenger counting system using head detection based on deep learning," *Journal of Information Processing Systems*, vol. 18, no. 3, pp. 428–442, 2022.

[12] H. Kim, S.-H. Lee, and M.-K. Sohn, "Real-time head detection for automated passenger counting in embedded systems," in *Proceedings of the 2019 3rd International Symposium on Computer Science and Intelligent Control*, pp. 1–5, 2019.

[13] D. P. Naranjo Valero *et al.*, "Tiempos de ascenso y descenso de los buses de acuerdo al comportamiento de los usuarios en las estaciones típicas de transmilenio," 2015.

[14] D. Jaramillo-Ramírez, W. D. Moreno Rendón, C. Burgos Anillo, and H. Carrillo, "Passenger counting in mass public transport systems using computer vision and deep learning." https://doi.org/10.17605/OSF.IO/WQAV3, 2023.

[15] D. Ttito, R. Quispe, A. R. Rivera, and H. Pedrini, "Where are the People? A Multi-Stream Convolutional Neural Network for Crowd Counting via Density Map from Complex Images," in *2019 International Conference on Systems, Signals and Image Processing (IWSSIP)*, pp. 241–246, 2019.

[16] V. Sindagi, "Crowdcount-mcnn." https://github.com/svishwa/crowdcount-mcnn, 2017. Last accessed 07 July 2021.

[17] D. Verona, "deep-crowd-counting_crowdnet." https://github.com/davideverona/deep-crowd-counting_crowdnet, 2016. Last accessed 07 July 2021.

[18] Y. Zhang, D. Zhou, S. Chen, S. Gao, and Y. Ma, "Single-Image Crowd Counting via Multi-Column Convolutional Neural Network," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 589–597, IEEE, jun 2016.

William David Moreno Rendón Electronic Engineer from Pontificia Universidad Javeriana (Bogotá, Colombia), with academic experience in artificial intelligence, signal processing and design of software and hardware solutions. He has worked as a cybersecurity analyst in the IT Security area at BTG Pactual Colombia. He is currently a researcher at the Department of Electrical and Computer Engineering at the University of Delaware, USA.

Carolina Burgos Anillo Electronic Engineer from Pontificia Universidad Javeriana (Bogotá, Colombia), with academic experience in artificial intelligence techniques and the design of hardware and software solutions. Since January 2022 she has been working in software engineering, focusing mainly on web development.

Daniel Jaramillo-Ramirez Electronic Engineer (UPB Medellin 2006), Master in Electronics (Uniandes Bogota 2008) and Ph.D. in Telecommunications (Supélec Gif-sur-Yvette, 2014). He has worked for Orange Labs (Paris) in research for 3GPP RAN1 standardization. Since 2014 he is an Assistant Professor in the Department of Electronics at Pontificia Universidad Javeriana in Bogota and is an active researcher on wireless communications, and urban transport, especially in quality of service for public transport systems and electric bicycles.

Henry Carrillo MSc. and Ph.D. in Computer Science and Systems Engineering from the University of Zaragoza (Zaragoza, Spain), Master in Electronic Engineering from the Pontificia Universidad Javeriana (Bogotá, Colombia) and Electronic Engineer from the Universidad del Norte (Barranquilla, Colombia), with experience in artificial intelligence techniques, the design of electronic hardware and algorithms for autonomous systems, including computer vision systems, mobile robotic systems, embedded systems, and intelligent systems.

Passenger Counting in Mass Public Transport Systems using Computer Vision and Deep Learning

William David Moreno Rendón, Carolina Burgos Anillo, Daniel Jaramillo-Ramirez, *Member, IEEE*, , and Henry Carrillo

***Abstract*—Estimating the number of people in vehicles and stations in public transport systems is crucial to improving the service quality. The TransMilenio system in Bogotá has serious drawbacks related to the lack of information in congestion situations. In this work, we present a computer vision method that uses deep learning techniques to estimate the number of people in TransMilenio's boarding platforms. We release the TransMilenio-Javeriana dataset with nearly 900,000 head labels on buses and stations. From these images, a deep learning architecture tuned for crowd counting was trained to generate density maps around the heads in the scene. Several head count methods were evaluated on the density maps. After testing these methods with 10,800 images, the results show a mean absolute error of 1 head per frame, equivalent to 11% relative error. The accuracy found is much better than its manual counterpart. The proposed system is also scalable and low-cost, which indicates that it has great potential to provide information for the planning and operation of public transport systems.**

***Index Terms*—Automatic passenger counting, station boarding, public transport, convolutional neural networks, computer vision.**

I. Introduction

Bogota is a capital city with a metropolitan population of 10 million inhabitants and a very high population density close to 24,000 inhabitants per square kilometer. It does not have a rail network for passenger transportation, and its primary means of mass transportation is a bus rapid transit network (BRT) known as TransMilenio (TM). The TM system, inaugurated in 2000, is recognized worldwide for sparking the modern BRT boom and has been replicated in hundreds of cities. This system, considered the BRT with the highest capacity in the world, moved close to 2.5 million passengers a day until before the pandemic, reaching capacities like those of heavy rail systems (between 20 and 50 thousand passengers per hour per direction), at a fraction of infrastructure and operating costs. However, to achieve such high capacities, TM requires intensive use of express services (routes that do not stop at all stations). The 114 km of roadways (dedicated lanes and stations with passing lanes) are covered by 8 less-used regular services (that stop at all stations) and some 90 express services that use multiple boarding platforms at each of the 147 stations in the system. In each station, all the boarding platforms and their gates are shared by 2, 3, or 4 different routes, making it impossible to queue, generating passenger congestion, shoving in boarding, and a sharp deterioration in the quality of service. Getting on the most congested services at peak hours can become physically impossible. Additionally, the high number of express services creates high uncertainty in the system, making its planning and operation difficult. It is more challenging to estimate bus occupancy, station congestion, waiting times, and internal transfers, making it difficult to take actions to improve service quality.

Computer vision is one of the technologies that could reduce the lack of information and improve the quality of service in critical situations of congestion in TM. A computer vision system with a medium computing capacity at the edge can identify heads, count people, estimate densities, track movement, and complete flow counting on buses and stations. Different computer vision techniques have been used in similar situations (see section I-A); however, there are no known works in the literature to estimate the number of people in high-capacity transportation systems, in high-density scenes, and in stations with boarding platforms shared by several routes. This research is also applied to similar situations that can occur in many mass transportation systems, including high-capacity rail systems, where counting people, estimating densities or waiting times is also of great importance.

A. State of the Art

Traditionally, human visual counts have been used to estimate the number of passengers. More recently, Automatic Passenger Counting (APC) methods have also been used, based on various data acquisition technologies, including carpet sensors, infrared sensors, weight sensors, different computer vision techniques, and systems that take advantage of the use of Wi-Fi devices [1]–[3].

The use of load sensors makes it possible to count people indirectly. Thanks to the weight data adjusted according to the action of braking systems, the distribution of passengers can be statistically characterized, as implemented since 2013 in the Copenhagen urban rail network. In this city, a comparison of a manual method with the implemented APC was made, and a high correlation (0.9) was obtained between both. The errors in the estimation are inherent to the variance of the weight of each person and the use of additional load (suitcases, bicycles, etc) [4].

People counting using infrared sensors is frequently performed in stores and buildings to have a more efficient flow control of entry and exit of people. In public transport systems, its use is limited to systems with low capacity or no congestion, such as in [2]. A good example of counting methods using

William David Moreno Rendón, Carolina Burgos Anillo, Daniel Jaramillo-Ramirez and Henry Carrillo are with the Department of Electronics, Pontificia Universidad Javeriana e-mail: moreno_william, carolinaburgos, d-jaramillo and h.carrillo @javeriana.edu.co .

Wi-Fi is the iABACUS [1] system. This system recognizes the MAC addresses of the devices within the range of an access point located within the bus without needing the devices to connect. It would also allow the device to be tracked if it moves within the transport system. The randomization of MAC addresses, a widespread trend in all phone manufacturers, strongly hinders the effectiveness of this technique.

Counting passengers using computer vision has also gained strength in recent decades, and different methods have been proposed. In general, they can be divided into three approaches [5]. The first method is counting based on *trajectory clustering*; this method is based on the hypothesis that the trajectories followed by the same human body are more similar to each other than those of other individuals. Visual characteristics are detected, and their trajectories are grouped into *clusters*, estimating the number of people via the number of *clusters*. A second method is regression-based counting; in this method, the number of people is estimated by learning the regression function between the characteristics of the input images and the people counted in it. A third method is called detection counting, in which an existing detector is selected or designed to detect people in the input image [3], [6]–[9]. In particular, the present work focuses on using a head detection technique to count people in situations where crowds could occur. Therefore, an application of *Deep Learning* for *Crowd Counting* in mass transportation systems is developed.

The previously mentioned works focus mainly on the problem of passenger flow, or the counting of moving passengers, inside buses or trains, mainly at the entry or exit of the vehicle. Some of these works focus on the adequacy of algorithms and neural networks for implementation in real-time [10]. Others are even trained with images of crowds that do not correspond to scenes of mass transportation or with images at angles susceptible to occlusion [11], [12]. This work differs from the available literature mainly for three reasons:

- Passengers are not followed using *tracking* algorithms; therefore, boardings or descents are not counted. Head counting and detection are only done on a still image, *frame by frame*.
- The images used are not inside vehicles, where the height of the camera and the space conditions generate images with few heads to count, and the flows are one passenger at a time. The images of this work are taken in a zenithal position from the roof of the boarding platforms, managing to capture up to 30 heads in the image at a time.
- This work includes the publication of a video repository with more than 200 hours of recordings, with 36,000 frames and close to 900,000 annotated heads, which should serve for the development of tracking and counting systems in any public bus transportation system or trains.

Therefore it includes the stages of acquisition (data repository), annotation and pre-processing, training, detection and head counting, and testing. The organization of this article is explained below. In section II the details of implementing a computer vision system in a TM station, its capacity, and its technical restrictions are described. In addition, the TM-PUJ video repository that contains the recordings obtained is also described. This repository remains open to the research community. The section III describes the pre-processing performed on the images before entering the information into the neural network. The section IV explains the neural network used, as well as the training and validation process. The detection and counting methods are exposed in the section V. In section VI, the results obtained are explained to finally draw conclusions in Section VII.

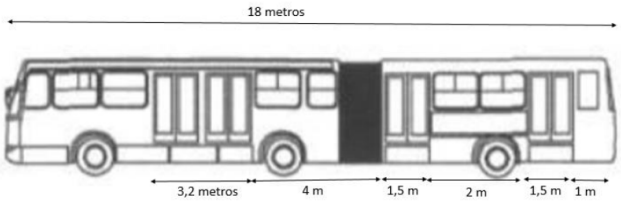

Fig. 1. Dimensions of an articulated bus, taken from [13].

II. Description of the Video Repository

A. Information Capture

In the traditional configuration of the TM system (until 2019), almost all stations included boarding platforms for articulated buses (150 passengers), making up most of the fleet. Few stations could receive bi-articulated buses (250 passengers). The articulated buses (18 m long) use platforms with 3 doors: a double door (3.2 m) in the front car and two single doors (1.5 m each) in the rear car, as shown in Figure 1.

Depending on the height of the station ceiling and the available viewing angles, a camera was installed on every single door and two cameras on the double door, completing 4 ELP-170 monocular cameras, whose distribution is observed in Figure 2. Each camera is connected to a Raspberry Pi B+ mini-computer with a 64 GB SD card, which depending on the brightness and movement in the image, could record compressed video for approximately 100 hours. To avoid very large files that are not easy to manipulate, the Raspberry cut the recording every 15 minutes or 27,000 frames and restarted with a new file (the video records at approximately 30 fps).

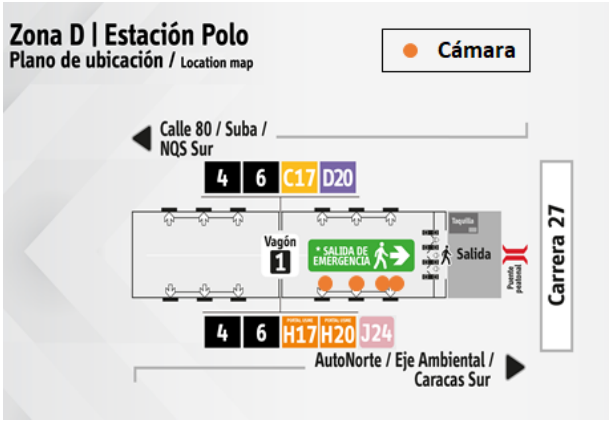

Fig. 2. Plan of the Polo station in 2019 and location of the cameras, taken from TransMilenio's website.

TABLE I
SUMMARY OF THE TRANSMILENIO-JAVERIANA VIDEO REPOSITORY AND THE FILES USED IN THIS STUDY.

Complete video repository					
Location		Recording hours	# Video-clips (15 min)		
Bus		120	480		
Station		107	430		
Presented study (used frames)					
Videoclips (1 min)	# Labeled heads	# Labeled frames	Training	Validation	Test
20	895231	36000	22660 (63%)	2520 (7%)	10800 (30%)

The devices made recordings between Wednesday, May 23, and Tuesday, May 29, 2019. They were installed at the Polo station on the Calle 80 corridor, in the west-east direction, on the first four eastern doors, which provided access to the H20 services, H17 and J24. The clips obtained, lasting 15 minutes, are part of the video repository called TransMilenio-Javeriana.

B. Structuring the Video Repository

The TransMilenio-Javeriana [14] repository comprises videos on buses and stations. Contains 120 hours of video on a bus with different levels of lighting and occlusion, including 4 different perspectives: a camera located to the left of a double bus door, another located to the right, another in the middle of the middle door, and finally a camera in the center of the rear door. In total, 480 video clips were obtained on the bus. In addition, the repository contains 107 hours of video at stations with different levels of lighting and occlusion, comprised of 3 perspectives of the door: one where the camera is to the left of the double door, another with the camera to the right, and finally the camera filming a simple door from the center. All cameras have a top view (zenithal). 430 videos were obtained at the station. Adding the station and the bus, 227 hours were recorded in 910 videos with an average duration of 15 minutes each.

For this investigation, 20 video clips were used in the station; each one has 1800 frames. The total number of images studied and the details of the repository are summarized in Table I.

To facilitate the analysis, improve the effectiveness of the counting methods, and review in which type of scenes the algorithm is most effective, 4 occupancy levels are used. The qualitative description of each level is presented below:

- **Level 1:** Low occupation. The station is almost empty. Images that record a maximum of 3 people.
- **Level 2:** Medium-low occupancy. Station with little influx of people. Images from 4 to 9 people.
- **Level 3:** Medium-high occupancy. Station with images between 10 and 25 people.
- **Level 4:** High occupancy. Station with complete obstruction of the passage in the doors and difficulty to move in the opening and closing of doors. More than 25 people per image.

The distribution of the test frames according to their occupancy level is shown in Fig. 3, where a relative balance among all levels can be seen.

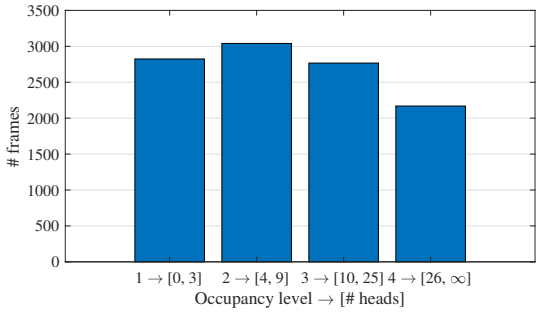

Fig. 3. Histogram of the number of frames vs. occupancy level for the 10800 test frames.

III. PRE-PROCESSING

The acquired images must be labeled and processed to enter the neural network at this stage. To complete the labeling of the 20 defined video clips, the CVAT program was used, which helps the user label consecutive frames, suggesting the location of the labels. In this way, 36,000 individual annotated images were completed, containing 895,231 heads.

A. Generation of Training Data

Crowd counting seeks to count the number of people in a crowded scene. One of the methods to perform this task is density estimation, which aims to convert a crowded image into its corresponding density map used to estimate the number of people in the image. [15] presents a neural network that estimates a density map of the location of heads in a crowd. This model learns from a series of kernels, as seen in Fig. 4, where they identify the location of people's heads at different scales. Each head is converted to a two-dimensional normal distribution in the density map.

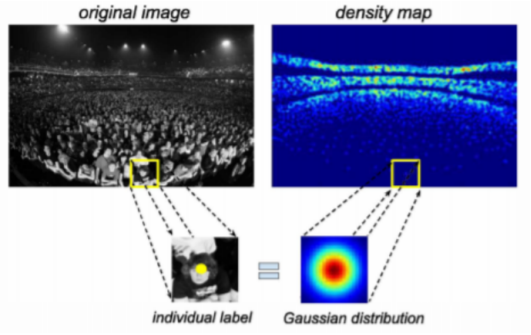

Fig. 4. Use of kernel in the original image to obtain a normal distribution in the density map as presented in [15].

This work used a two-dimensional Gaussian kernel described in the equation (1). The variables x and y determine the kernel position, while σ is the standard deviation of the Gaussian distribution, approximating the kernel's width. When

applying the Gaussian kernel, the image will be blurrier the larger the value of σ is.

$$G_\sigma(x, y; \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{1}$$

To use Gaussian kernels in two dimensions, it is necessary to maintain a compromise between the distortion of the filtered image and the precision with which the kernel indicates a head. Therefore, two alternatives were analyzed: fixed kernel (with fixed σ) and variable kernel.

1) Fixed Kernel: First, an adaptation of the map generation method used in the original implementation of [16] was made. A window size of 29x29 pixels and $\sigma = 8$ were used. The result is shown in the annotation of Figure 5 with an image of a station from our repository.

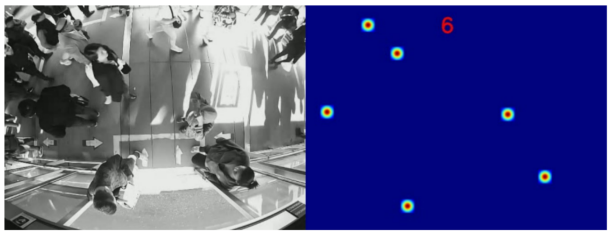

Fig. 5. Density map with a fixed kernel.

2) Variable Kernel: This option consists of using a Gaussian kernel over the center point of the annotations but, this time, changing the size and parameter σ of each kernel considering its nearest neighbors. To complete this function, the Gaussian filter implemented in [17] was used, using the distances to the nearest neighbors to calculate the value of σ for each kernel. Figure 6 shows the result with the annotations made.

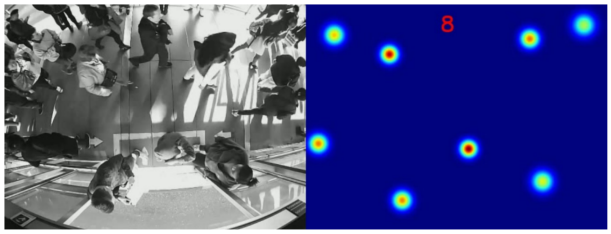

Fig. 6. Density map with a variable kernel.

Additionally, a region of interest (ROI) was defined on the images to eliminate the points near the edges where the heads are very small and sometimes indistinguishable, considering that they can generate noise in the data. In this case, the ROI is defined between pixels 25 and 615 in *x* and between pixels 40 and 465 in *y*, taking into account that the original images have a size of 640x480. Figure 9 shows the ROI limit as a dotted red line.

IV. Training

After evaluating various neural network configurations and their results in head counting and crowd counting applications (Yahiaoui 2010 [8], Lumentut 2015 [7], Zhang 2016 [18], Perng 2016 [6], Li 2016 [9], Sun 2019 [5], Labit-bonis 2021 [10], Kim 2022 [11], see Table III), the implementation made in [18] was chosen. It is a 5-layer multi-column convolutional neural network (MCNN), which implements filters in each column to fit the density maps corresponding to heads at different scales.

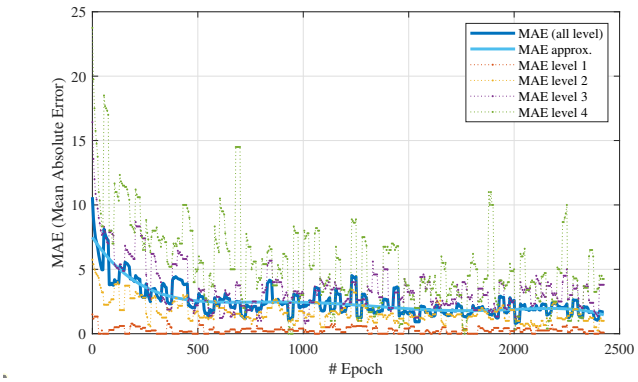

Fig. 7. Number of epochs vs MAE according to the level of occupancy in the image.

Considering that a deep learning network must be trained iteratively, it is necessary to have a loss function. The loss function traditionally used in regression problems is the mean square error (MSE), which can be seen in the equation (2). The mean absolute error (MAE) can also be used, as shown in the equation (3).

$$\text{MSE} = \frac{\sum_{i=0}^{n} \left(Y_i - \hat{Y}_i\right)^2}{n} \tag{2}$$

$$\text{MAE} = \frac{\sum_{i=0}^{n} \left|Y_i - \hat{Y}_i\right|}{n} \tag{3}$$

During training, a fine-tuning process was also completed by changing different hyper-parameters in the model. Using the MAE it was found that the density maps with variable kernel do not reduce the loss function. The best training result was obtained with the fixed kernel density maps. To estimate the number of heads in the image, the sum of all the pixels in the density map is made.

The following is a summary of the training and validation parameters obtained heuristically:

- **Scale factor for *Ground Truth* (GT):** The input density maps are multiplied by 1.1 to produce a higher intensity in the kernel and a lower MAE in training.
- **Batches:** The 22660 training frames are divided into 63 batches, avoiding overloading the computer's memory.
- **Epochs per batch:** 20 iterations are carried out on each batch, avoiding overflowing the processing capacity.
- **Learning rate:** The step size for each iteration seeks to find a minimum in the loss function. Its value was 0.9, found heuristically.
- **Total Epochs:** Number of iterations of the model that are performed during training, which in this case is 2450.

Figure 7 shows the trends of the MAE during training using the validation images. It should be noted that the MAE is used to measure learning performance based on the number of *epochs*. After about 2500 epochs, the network showed signs of overtraining.

V. Detection and Head Count

In order to obtain a better estimate of the number of people represented in the network output density maps, it is necessary to explore different counting methods for the scenes obtained and the different occupancy levels in each image.

A. Counting Methods

1) Integral counting: The sum of the pixel values of the grayscale image given by the equation (4), where $f(x_i)$ is the image density map of n pixels and x_i is the value of the image in each pixel.

$$\sum_{i=1}^{n} f(x_i) \tag{4}$$

2) Peak counting: Using the Otsu binarization method, a threshold is found for each image that determines those pixels that can represent peaks in the image. Then, a search for the maximum local peak is performed employing dilation of the image as indicated in the equation 5, where $f(x)$ is the original image and $f(x) \oplus Y$ the dilated image. Next, the pixels that meet this condition in the radius of a head are grouped into a single peak; avoiding that a single head contains several peaks and is counted more than once

$$(f(x) \oplus Y) == (f(x)) \tag{5}$$

All counting methods performed better at certain occupancy levels but were ineffective at each occupation level. Therefore, two new counting methods were defined, combining the previous methods as follows.

3) Combined method 1: The integral method is used for counts of less than 7 heads, and for counts of 8 or more heads, the peak count is used. In practice, both methods are executed in parallel and are chosen according to the value determined by both. In case of contradiction, the peak method is chosen.

4) Combined method 2: The same ranges are used as in the previous method, but for counts between 4 and 17 heads, pixels with a non-zero value (pixels representing possible heads) are counted and divided by the estimated number of pixels in each head.

B. Evaluation of Counting Methods

In order to evaluate the counting methods with better proportionality, the mean relative error, known as MRE, was used:

$$\mathrm{MRE} = \frac{100}{n} \sum_{i=0}^{n} \frac{\left|Y_i - \hat{Y}_i\right|}{Y_i}, \tag{6}$$

where Y_i is the reference value, $\hat{Y}_i$ is the result of the count and n the number of frames considered. When the Ground Truth (GT) is $Y_i = 0$ (about 10 % of the frames), to avoid division by 0, the MRE is replaced by the average value of the $rmMRE$ in those cases with the same MAE $|Y_i - \hat{Y}_i|$ but with $Y_i \neq 0$.

TABLE II
MAE and MRE results for the Integral counting method by occupancy levels in test frames.

Nivel de ocupación	Rango de cabezas	MAE	MRE (%)
1	0 - 3	0.22	10.6
2	4 - 9	0.96	16.9
3	10 - 25	1.90	11.2
4	> 25	2.83	9.3

In this way, performance can be evaluated in two dimensions: the absolute error (MAE) and the relative error (MRE), and determine the best method as the one that provides the lowest combination of both errors for all station occupancy levels.

A scatter plot is shown in Fig. 8 for the MAE and the MRE of the estimation of the number of heads from the 4 methods described above. Each one of the points corresponds to an average of the error on 108 different images, obtaining a total of 100 points for each method, thus covering all the 10,800 test images for the stations.

It is important to first note the different scales of the four graphs, both in the MAE and MRE. The absolute error increases with a higher occupancy level while the relative error decreases. The integral counting method achieves the best result for level 1 with a MAE of 0.22 and an MRE of 10.59%. At level 2, the integral count is also the best, with a MAE of 0.96, and an MRE of 16.9%. In the first two levels, the peak count has a high number of absolute and relative errors and the combined counts are closer to the integral count. At level 3, the integral count is much higher than the others with a MAE of 1.90 and an MRE of 11.23%, while the peak count drags the combined methods towards 4 errors per frame and 25% MRE. Finally, in the images of the last level, the result is observed in the lower right graph and shows that the integral count is slightly better than the other methods with a MAE of 2.83 and an MRE of 9.29%, this being the best method for the 4 levels presented in images in the station. The combined methods offered good results in networks trained with both station and bus images, but in this work, where the network was trained only with station images, the combined methods did not outperform the peak count method at any of the levels of occupation.

Taking into account the results provided by the different basic and combined count methods, the best result is obtained with the integral count method as shown by occupancy levels in Table II. Averaging the results at all levels, the integral method presents a MAE of 1.39, an MRE of 14.24% and an MSE of 2.07, indicating that the error in the count on the generated maps is almost always and regardless of the occupancy level between 1 and 2 heads.

To have a better understanding of the system's operation in visual terms, examples are shown in Figure 9. For previously annotated images, the original image is presented on the left

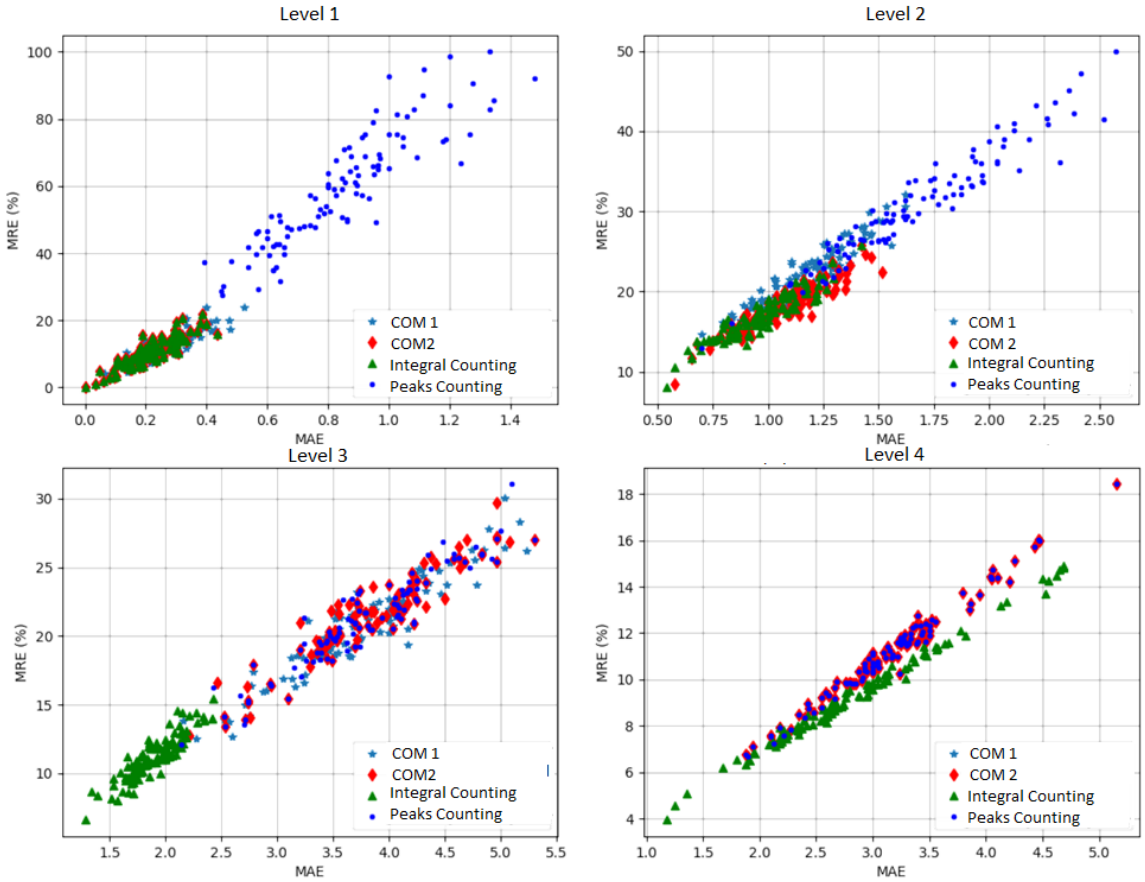

Fig. 8. Comparison of MAE and MRE (%) for each occupancy level, with all counting methods. Each point represents an average of 108 frames with the same occupancy level. In total there are 100 points in each image including all test frames.

along with the number of heads present in the annotations (AN), in the center the density map taken as *ground truth* (GT) with the count provided by making the integral on it, and on the right is the density map generated by the network superimposed with the original image to be able to visualize where the map locates the heads, accompanied by the estimate (ET) obtained with the count method. A video with some example results is also available for viewing in the repository attached to this article.

VI. Results and Analysis

For test video clips that are not annotated, the original image is presented on the left, the map delivered by the network overlaid with the image in the center, and the map displayed as a heat map along with the estimated value on the right. Since there are no annotations of these images, scenes where it is easy to visually count the people present were chosen. In Fig. 10, you can see individual images that are part of some of these videos.

After completing the training, validation, and finding the best counting method, the results obtained with the test images are presented: 10800 frames randomly selected from the 20 clips. The results show an inherent temporal dynamic. By arranging the frames sequentially, depending on the occlusion and illumination levels, some parts of the sequence can be more difficult to detect and count, leading to under-counting (false negatives) or over-counting (false positives). If they are simultaneous, both errors are compensated. Figure 11 shows how, despite the count being independent in each frame and not having any tracking method in the sequence, the result is very close to GT at all occupancy levels. Even the good

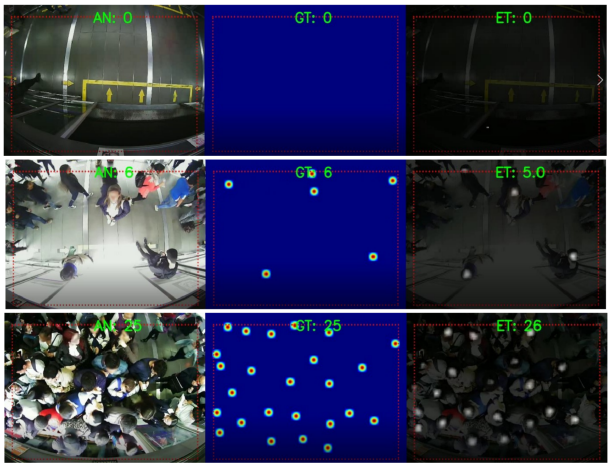

Fig. 9. Evaluation of the model with empty, medium and full station according to the value of GT (estimated count of *Ground Truth* or labeling) and ET (estimated value obtained with the integral counting method).

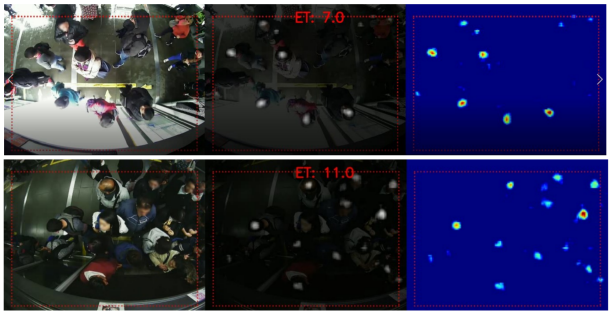

Fig. 10. Example of counting with images without annotating (without GT) in a station with little and medium congestion.

results are maintained when many people abruptly leave the image (after boarding a bus), as seen in the box "Clip 10, Level 2-3". Despite the good results, in that same sequence, the count does not register about 3 of the heads in the scene, generating a high MRE for that video clip.

After analyzing all the test image results in the 20 video clips, only one presented an atypical result and was therefore excluded from the analysis. Additionally, when reviewing Figure 11 in detail, the four upper graphs show how the counting signal shows rapid oscillations around a GT value that is often constant for many frames. To reduce these errors, a 3-position moving fashion filter was produced, which helps reduce the total error marginally in all video clips. Also, in the lower part of Figure 11, the histogram of the absolute error in each frame is observed for each occupancy level. 39% of the total frames have a perfect count (MAE is zero). Additionally, the need to reduce the error variance is confirmed, probably through the further training of the network, especially in clips whose predominant level is level 3.

Finally, the results of the test frames for the 19 video clips are observed in the scatter plot of Figure 12. The predominant level in a clip is called the rounded average of the occupancy level of each frame of the clip. The clips with predominant level 3 show the highest absolute error values but are always below 3. All the relative error values are below 25%, and only two of the clips exceed 16%. The average results over the total test frames show how the absolute error is around 1 head per frame, while the relative error is close to 11%. This implies that the proposed system manages to distinguish occupancy levels with very high effectiveness, and the error found is marginal for the planning and operation of a public transport system under similar conditions of high capacity.

A. Results Comparison

Finally, to put our work in context, the results obtained are compared with others from previous works focusing on similar situations. Table III summarizes similar works, with the different metrics considered by the respective authors: Accuracy, Precision and Recall, and the Mean Absolute Error (MAE). The result of this work is shown in the last line. The accuracy of 89% and MAE of 1.01 were obtained with 20 video clips from the TransMilenio-Javeriana repository adapting density maps and the MCNN convolutional network.

VII. Conclusions

Although the poor quality of service offered in mass public transportation is evident, it is difficult to focus efforts on the planning or operation of the system due to the lack of accurate information on passenger congestion in vehicles and stations. Given the difficulty of counting people by traditional methods, this paper presents a computer vision method that is effective, easy to implement, scalable, and low-cost.

Using a database of more than 100 hours of recording, and almost 900,000 tagged heads in 36,000 images (frames), a deep-learning neural network was trained and evaluated to estimate the number of people using maps of density with Gaussian kernels.

After evaluating four different counting methods, the integral count shows an average error of 1 head per frame, corresponding to a relative error of 11%, equivalent to an accuracy of 89%. This was obtained on a set of 10,800 test squares with four different occupancy levels.

The proven method has a much better counting capacity than traditional methods based on manual gauging, and it works in different lighting, occlusion, and congestion conditions. The accuracy found of 89%, with an average error of 1 head per frame, makes it possible to estimate very well occupancy ranges and densities in the areas of the field of view of the installed cameras, providing valuable information for both long-term system planning, as well as for its operation in the short term.

The results show a natural relationship between the number of heads in successive frames. This information, which has not been used in this work, could help reduce the average error found and make it possible to correctly estimate the flows of people in motion, to calculate, for example, the occupancy profile of vehicles in real-time.

References

[1] M. Nitti, F. Pinna, L. Pintor, V. Pilloni, and B. Barabino, "Iabacus: A Wi-Fi-based automatic bus passenger counting system," *Energies*, vol. 13, no. 6, 2020.

[2] A. Olivo, G. Maternini, and B. Barabino, "Empirical study on the accuracy and precision of automatic passenger counting in european bus services," *The Open Transportation Journal*, vol. 13, no. 1, 2019.

[3] I. Grgurević, K. Juršić, and V. Rajič, "Review of automatic passenger counting systems in public urban transport," in *5th EAI International Conference on Management of Manufacturing Systems*, pp. 1–15, Springer, 2022.

[4] B. F. Nielsen, L. Frølich, O. A. Nielsen, and D. Filges, "Estimating passenger numbers in trains using existing weighing capabilities," *Transportmetrica A: Transport Science*, vol. 10, pp. 502–517, jul 2014.

[5] S. Sun, N. Akhtar, H. Song, C. Zhang, J. Li, and A. Mian, "Benchmark Data and Method for Real-Time People Counting in Cluttered Scenes Using Depth Sensors," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, pp. 3599–3612, oct 2019.

[6] J.-W. Perng, T.-Y. Wang, Y.-W. Hsu, and B.-F. Wu, "The design and implementation of a vision-based people counting system in buses," in *2016 International conference on system science and engineering (ICSSE)*, pp. 1–3, IEEE, 2016.

[7] J. S. Lumentut, F. E. Gunawan, *et al.*, "Evaluation of recursive background subtraction algorithms for real-time passenger counting at bus rapid transit system," *Procedia Computer Science*, vol. 59, pp. 445–453, 2015.

[8] T. Yahiaoui, L. Khoudour, and C. Meurie, "Real-time passenger counting in buses using dense stereovision," *Journal of Electronic Imaging*, vol. 19, no. 3, p. 031202, 2010.

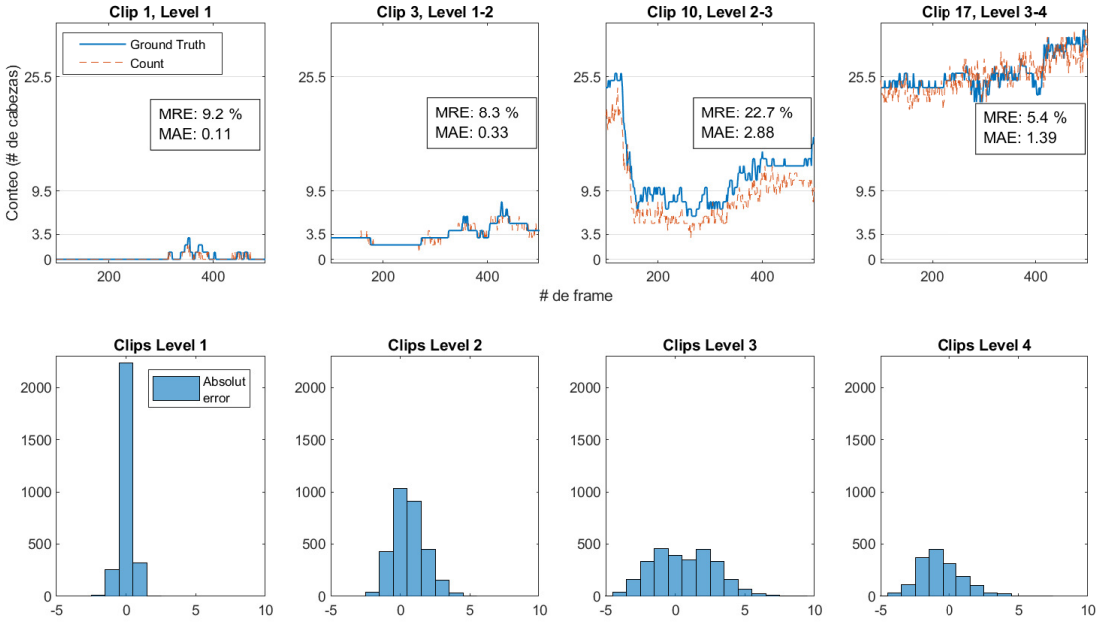

Fig. 11. Top: Examples of video-clip counts with all occupancy levels with their respective MAE and MRE. Bottom: histograms of the absolute error for all the frames according to their occupancy level.

TABLE III
COMPARATIVE RESULTS OF DIFFERENT METHODS FOR HEAD COUNTING ADAPTED FOR PUBLIC TRANSPORT.

Paper	Situation	Method	Result
Yahiaoui 2010 [8]	Flow inside bus,	Stereovision. Segmentation, binarization, tracking	Accuracy 97%
Lumentut 2015 [7]	Flow in station	Counting (AMF)	Recall 88%, Precision 19%
Zhang 2016 [18]	Crowd counting	Counting (CNN)	MAE 1.60
Perng 2016 [6]	Flow inside bus	Background subst., detection, tracking	Accuracy 87%
Li 2016 [9]	Flow inside bus	RGB+D. Detection and tracking	Accuracy 92%
Sun 2019 [5]	Flow inside bus	RGB+D. 3D body projection	Precision and Recall (92%)
Labit-bonis 2021 [10]	Flow inside bus	YOLOv5 + DeepSORT	Accuracy 95%
Kim 2022 [11]	Flow inside bus	Real-time YOLOv3	Accuracy 99%
Moreno 2022	Counting in station	Density maps, MCNN	Accuracy 89% MAE 1.01

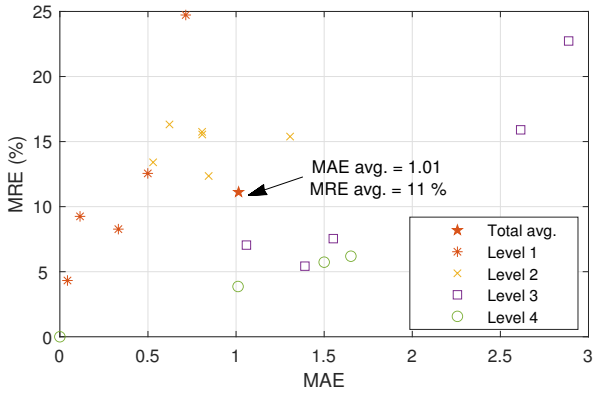

Fig. 12. MAE vs MRE for the test frames in the 19 video clips analyzed, according to their dominant occupancy level.

[9] F. Li, F. Yang, H. Liang, and W. Yang, "Automatic passenger counting system for bus based on rgb-d video," in *2nd Annual International Conference on Electronics, Electrical Engineering and Information Science, EEEIS*, 2016.

[10] C. Labit-Bonis, J. Thomas, and F. Lerasle, "Visual and automatic bus passenger counting based on a deep tracking-by-detection system." working paper or preprint, Oct. 2021.

[11] H. Kim, M.-K. Sohn, and S.-H. Lee, "Development of a real-time automatic passenger counting system using head detection based on deep learning," *Journal of Information Processing Systems*, vol. 18, no. 3, pp. 428–442, 2022.

[12] H. Kim, S.-H. Lee, and M.-K. Sohn, "Real-time head detection for automated passenger counting in embedded systems," in *Proceedings of the 2019 3rd International Symposium on Computer Science and Intelligent Control*, pp. 1–5, 2019.

[13] D. P. Naranjo Valero *et al.*, "Tiempos de ascenso y descenso de los buses de acuerdo al comportamiento de los usuarios en las estaciones típicas de transmilenio," 2015.

[14] D. Jaramillo-Ramírez, W. D. Moreno Rendón, C. Burgos Anillo, and H. Carrillo, "Passenger counting in mass public transport systems using computer vision and deep learning." https://doi.org/10.17605/OSF.IO/WQAV3, 2023.

[15] D. Ttito, R. Quispe, A. R. Rivera, and H. Pedrini, "Where are the People? A Multi-Stream Convolutional Neural Network for Crowd Counting via Density Map from Complex Images," in *2019 International Conference on Systems, Signals and Image Processing (IWSSIP)*, pp. 241–246, 2019.

[16] V. Sindagi, "Crowdcount-mcnn." https://github.com/svishwa/crowdcount-mcnn, 2017. Last accessed 07 July 2021.

[17] D. Verona, "deep-crowd-counting_crowdnet." https://github.com/

davideverona/deep-crowd-counting_crowdnet, 2016. Last accessed 07 July 2021.

[18] Y. Zhang, D. Zhou, S. Chen, S. Gao, and Y. Ma, "Single-Image Crowd Counting via Multi-Column Convolutional Neural Network," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 589–597, IEEE, jun 2016.

William David Moreno Rendón Electronic Engineer from Pontificia Universidad Javeriana (Bogotá, Colombia), with academic experience in artificial intelligence, signal processing and design of software and hardware solutions. He has worked as a cybersecurity analyst in the IT Security area at BTG Pactual Colombia. He is currently a researcher at the Department of Electrical and Computer Engineering at the University of Delaware, USA.

Carolina Burgos Anillo Electronic Engineer from Pontificia Universidad Javeriana (Bogotá, Colombia), with academic experience in artificial intelligence techniques and the design of hardware and software solutions. Since January 2022 she has been working in software engineering, focusing mainly on web development.

Daniel Jaramillo-Ramirez Electronic Engineer (UPB Medellin 2006), Master in Electronics (Uniandes Bogota 2008) and Ph.D. in Telecommunications (Supélec Gif-sur-Yvette, 2014). He has worked for Orange Labs (Paris) in research for 3GPP RAN1 standardization. Since 2014 he is an Assistant Professor in the Department of Electronics at Pontificia Universidad Javeriana in Bogota and is an active researcher on wireless communications, and urban transport, especially in quality of service for public transport systems and electric bicycles.

Henry Carrillo MSc. and Ph.D. in Computer Science and Systems Engineering from the University of Zaragoza (Zaragoza, Spain), Master in Electronic Engineering from the Pontificia Universidad Javeriana (Bogotá, Colombia) and Electronic Engineer from the Universidad del Norte (Barranquilla, Colombia), with experience in artificial intelligence techniques, the design of electronic hardware and algorithms for autonomous systems, including computer vision systems, mobile robotic systems, embedded systems, and intelligent systems.