

Bridge Crane Monitoring using a 3D LiDAR and Deep Learning

Jesús M. García, Jorge L. Martínez, Senior Member, IEEE and Antonio J. Reina

Abstract—The use of overhead cranes in warehouses and factories has advantages for handling and transporting bulky and/or heavy loads. But it also involves risks such as collisions with other fixed or mobile elements in the working environment. Different types of sensors have been used for monitoring its operation, mainly artificial vision. In this paper, it is employed a three-dimensional (3D) LiDAR to capture the workspace of a bridge crane. The point clouds generated by this laser sensor are delivered to a convolutional neural network to detect the position of the bridge and its carriage, which allows to locate the hook and the suspended load afterwards. Additionally, the laser scans can also be used to warn the operator of possible collisions with fixed elements of the warehouse. The tests carried out show that the proposed system can be successfully used for monitoring overhead cranes.

Index Terms—Bridge crane, Collision detection, Convolutional neural network, Deep learning, 3D LiDAR.

I. INTRODUCCIÓN

El puente grúa es un tipo específico de grúa embebida en el interior de fábricas y almacenes que se utiliza para izar y desplazar cargas pesadas y/o de gran porte. Normalmente el puente grúa está formado por un puente y un carro que permiten el movimiento horizontal en dos direcciones, mientras que sobre el carro hay un sistema de polipastos con gancho que permite la elevación vertical de la carga. Las ventajas de los puentes grúa son evidentes, pero también presentan retos relevantes, ya que, en ocasiones, resulta difícil asegurar el movimiento de la carga, lo que puede provocar costosos accidentes. Los puentes grúas son normalmente teleoperados por personas por medio de un mando a distancia. Para automatizarlos, se requiere incorporar sensores adecuados que permitan conocer en tiempo real la posición de la carga respecto del entorno de trabajo.

Para la mejora de estos equipos se han estudiado diferentes diseños de manera que se cumplan con las necesidades específicas que atenderán [1][2]. De hecho, el proceso de mejora del hardware involucrado es continuo para dotar a estos equipos con sensores y controladores que permitan a la grúa ser más precisa y segura evitando situaciones peligrosas como la colisión con otros elementos fijos o móviles [3][4].

J. M. García is with the Laboratorio de Prototipos, Universidad Nacional Experimental del Táchira, Av. Universidad, sector Paramillo, San Cristóbal, Venezuela (e-mail: jmgarcia@unet.edu.ve).

J. L. Martínez and A. J. Reina are with the Universidad de Málaga, Andalucía Tech, Dpto. Ingeniería de Sistemas y Automática, 29071- Málaga, España (email: jlmartinez; ajreina@uma.es).

Para asegurar el movimiento sin oscilaciones y la precisión de posicionamiento al trasladar cargas se han planteado sistemas que utilizan múltiples cables actuados para que, combinando la tensión entre ellos pueda disminuirse la oscilación del gancho [5]. Otras soluciones incluyen sistemas de control predictivo [6] y algoritmos de inteligencia artificial que contienen redes neuronales, lógica difusa y algoritmos genéticos [7]. Pero la implementación de estos algoritmos necesita de entradas reales tales como el ángulo de oscilación del cable, el cual se puede obtener mediante visión artificial [8][9]. Adicionalmente, se pueden considerar otros factores tales como la velocidad del viento [5] o el carácter flexible de los cables que sostienen la carga [10].

Otro reto a considerar en los puentes grúa está en la planificación de trayectorias seguras para el movimiento de la carga. Para ello se parte de un mapa del entorno de trabajo, que se elabora normalmente usando sistemas de visión artificial [8][11][12]. La planificación de trayectorias se define a través de diferentes algoritmos, entre ellos: campos potenciales [11][13] y exploración rápida de árbol aleatorio con optimización utilizando enjambres de partículas [14]. Pero esta planificación también debe tomar en consideración que el espacio de trabajo es dinámico y pueden aparecer obstáculos inesperados [15].

Además, se han desarrollado sistemas de evitación de obstáculos que se realimentan en tiempo real utilizando sistemas de visión [16] y de ultrasonidos [17]. Así, en [18] se definió la distancia mínima segura entre la carga y cualquier objeto, que toma en consideración factores importantes como las oscilaciones de la carga y su velocidad de desplazamiento.

El reconocimiento de los objetos transportados en puentes grúa se ha logrado utilizando sistemas inteligentes como redes neuronales alimentadas con información geométrica (vértices, por lo general) obtenida de sistemas de visión [19]. Asimismo, para que trabajen coordinados dos puentes grúa y puedan mejorar la productividad, se han integrado sus sistemas de control y monitorización [20].

Todas las áreas de investigación descritas previamente tienen un denominador común: requieren la adquisición de información del entorno de trabajo como entrada para los diferentes sistemas de control que automatizan el proceso. Además de cámaras, se han utilizado sensores tales como ultrasonidos, telémetros láser 1D y 2D, codificadores y elementos de final de carrera [3][4][7][8]. Extraer información relevante requiere tiempo de cómputo que, en algunos casos, como en el reconocimiento de obstáculos móviles, es limitado.

Recientemente han aparecido los telémetros láser 3D LiDAR

(*Light Detection and Ranging*). Estos sensores pueden ser bastante útiles, ya que resulta factible adquirir nubes de puntos del entorno de trabajo. Esta información se puede actualizar con una frecuencia elevada y es más fácil de analizar que las imágenes. La tecnología LiDAR puede utilizarse en diversas aplicaciones como navegación, robótica, detección remota y asistencia avanzada para sistemas de conducción [21].

Por otra parte, el aprendizaje profundo (*deep learning*) permite a una computadora aprender a reconocer características complejas en los niveles superiores de una jerarquía a través de la composición de características más simples procedentes de los niveles inferiores de dicha jerarquía [22]. Una de las herramientas frecuentemente utilizadas en aprendizaje profundo son las Redes Neuronales Convolucionales (CNN, *Convolutional Neural Network*) en tareas tales como el reconocimiento de patrones en imágenes, vídeo, audio o nubes de puntos. Normalmente, están compuestas por capas convolucionales y submuestreo no conectadas, seguidas por capas totalmente conectadas que realizan la clasificación final [23][24].

En conjunto, LiDARs y CNNs son utilizadas actualmente en múltiples aplicaciones, principalmente en aquellas relacionadas con la detección de diversos elementos en sistemas de conducción autónomos. Por ejemplo, en [25] se utilizan para detectar otros vehículos en movimiento; en [26] se utilizan junto con una cámara, para la detección de obstáculos fijos con formas predefinidas (conos de tránsito) que delimitan el espacio de circulación de los vehículos; y en [27] se utilizan un LiDAR para la detección de peatones, aunque en este caso está acompañado de una máquina de soporte vectorial.

Respecto al uso de estas herramientas para la automatización de puentes grúas, son pocos los trabajos desarrollados en este caso. En [28] se reportó el uso de un LiDAR para medir el espacio que sería utilizado por el puente grúa y para segmentar objetos de la nube de puntos; y en [29] se utilizaron varios LiDARs junto con una DGCNN (*Dynamic Graph CNN*) para segmentar y etiquetar las distintas partes de una grúa a partir de una nube de puntos 3D. Esto como punto de partida para una futura detección del gancho y la creación de un sistema para la evitación de obstáculos.

En este trabajo se propone el uso de un LiDAR 3D para la monitorización de un puente grúa; este sensor, en conjunción con herramientas de aprendizaje profundo, sirve para obtener la posición del gancho durante todo el desplazamiento del puente grúa (cuando no se transporta carga) o la posición de la carga suspendida. Adicionalmente, se estimó el volumen aproximado que aropa la carga y se analizó su entorno para detectar posibles colisiones y la emisión de señales de alarma.

El resto del artículo está organizado de la siguiente forma. En la Sección 2 se presentan tanto el puente grúa como el sensor utilizado para su monitorización. En la Sección 3 se describe la detección del puente y del carro mediante aprendizaje profundo. En la Sección 4 se procede con la localización del gancho y de la carga mediante agrupamiento de puntos 3D. En la Sección 5 se proponen dos procedimientos para detectar riesgos de colisión. En la Sección 6 se presentan los resultados experimentales con una aplicación Matlab desarrollada

específicamente. En la última sección se sintetizan las conclusiones y oportunidades de investigación en esta área.

II. SISTEMA EXPERIMENTAL DE MONITORIZACIÓN

El caso de estudio de este trabajo se corresponde con un puente grúa de tipo pórtico (ver Fig. 1a), el cual posee tres grados de libertad:

- Movimiento del puente a lo largo de las guías.
- Desplazamiento del carro sobre el puente.
- Movimiento vertical del gancho respecto del carro.

El puente posee una amplitud de 10 m y sirve para transportar cargas en una nave taller de 20 m de largo en la Escuela de Ingenierías Industriales de la Universidad de Málaga.

Se ha utilizado un LiDAR 3D Velodyne VLP-16 cuyas especificaciones se detallan en la Tabla I. Este equipo fue posicionado 1.4 m por debajo del puente y a 5.3 m del suelo, en una localización que le permite adquirir datos provenientes del espacio de trabajo incluyendo el puente y el carro (ver Fig. 1b).

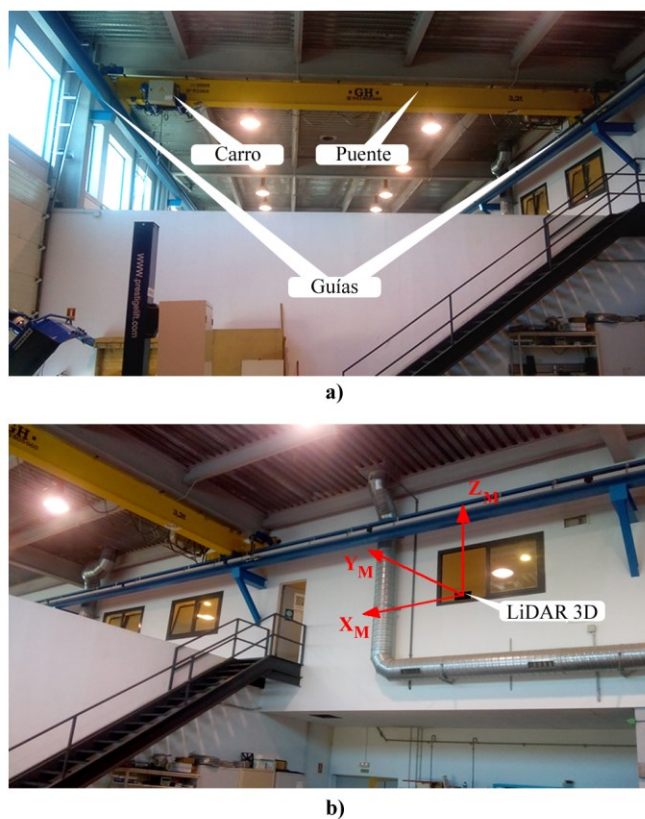


Fig. 1. Elementos del puente grúa (a) y colocación del LiDAR 3D en la nave taller (b).

Adicionalmente, se definió un sistema de referencia fijo, denominado $X_M Y_M Z_M$, cuyo origen coincide con la posición del LiDAR 3D, de manera que la dirección X_M corresponde con la dirección de avance del puente, la dirección Y_M es paralela a la dirección de movimiento del carro y el eje Z_M coincide con la dirección vertical (ver Fig. 1b).

Se utilizó el LiDAR a la menor frecuencia de adquisición posible de barridos láser 3D (5 Hz), la cual permite obtener la mejor resolución angular horizontal del mismo de 0.1° .

TABLA I
ESPECIFICACIONES DEL LIDAR 3D VELODYNE VLP-16

Número de rayos láser	16
Rango de medición	100 m
Precisión	± 3 cm
Campo de visión horizontal	360°
Campo de visión vertical	30° (-15° a $+15^\circ$)
Resolución angular vertical	2°
Resolución angular horizontal	$0.1^\circ - 0.4^\circ$
Frecuencia de adquisición de barridos	5 - 20 Hz

III. DETECCIÓN DEL PUENTE Y DEL CARRO USANDO APRENDIZAJE PROFUNDO

Las primeras pruebas de adquisición de barridos 3D permitieron comprobar la reducida cantidad de puntos que se captaban del gancho (ver Fig. 2), a pesar de tener éste un tamaño apreciable (0.5 m de alto x 0.21 m de ancho). Ello se debe a la insuficiente resolución angular vertical del LiDAR 3D de 2° (separación entre dos rayos láser consecutivos).

Por ello, se decidió realizar la detección del puente y del carro como primer paso debido a sus mayores dimensiones, que los hace más fáciles de identificar en un barrido láser 3D. Una vez identificados estos elementos, se podrían inferir las coordenadas del gancho y de la carga situadas por debajo del carro.

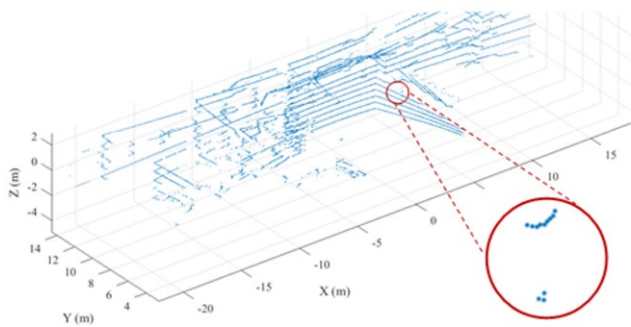


Fig. 2. Detalle de los puntos captados del gancho en un barrido láser 3D.

Esta detección se realizó utilizando una CNN provista por Matlab denominada PointNet++ [30], la cual está diseñada para extraer características de alto nivel dentro de nubes de puntos 3D a partir de la detección de características geométricas locales formadas por puntos vecinos. En particular, la red utilizada posee 79 capas distribuidas entre entrada de datos, convolución, normalización, muestreo, agrupamiento, interpolación y clasificación. Las capas de esta CNN están dispuestas de acuerdo a una arquitectura DAG (*Directed Acyclic Graph*) donde las capas pueden tener entradas o salidas en otras capas [31].

El objetivo de la CNN es clasificar los puntos de un barrido láser en tres categorías: puente, carro y resto de puntos. Para su entrenamiento se utilizaron nubes sucesivas de puntos que fueron acondicionadas de la manera que se describe a continuación. En primer lugar, se limpió cada nube para eliminar rangos inválidos y puntos alejados fuera de la nave a través de las ventanas (ver Fig. 3a). A continuación, se

seleccionaron aquellos puntos que cumplieran con la restricción $1 \leq z \leq 2$ m, ya que a esta altura está posicionado tanto el puente como el carro (ver Fig. 3b), quedando el gancho fuera con $z < 1$. En tercer lugar, la nube resultante, de poco más de 2500 puntos, es muestreada de forma aleatoria para fijar el número de puntos en 2500, lo cual define el número fijo de entradas que poseerá la CNN a entrenar.

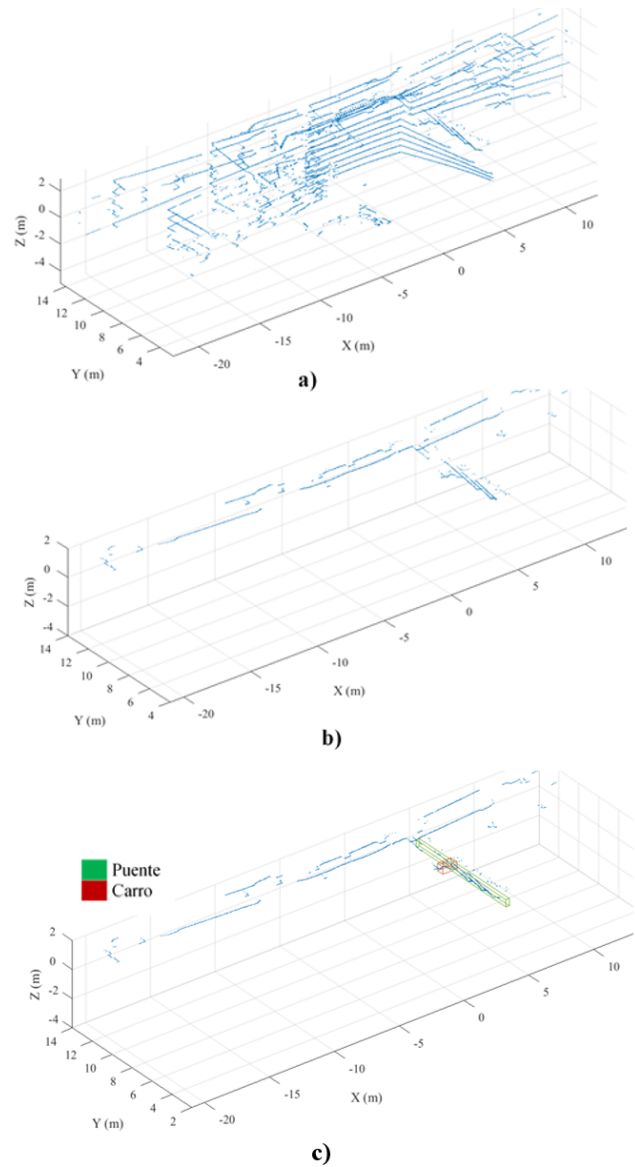


Fig. 3. Procesamiento de una nube de puntos 3D (a), puntos para el entrenamiento de la CNN (b) y regiones de interés sobreimpresas (c).

Posteriormente, fueron etiquetados todos los puntos de cada barrido láser de una secuencia. Para ello, se utilizó la herramienta *LidarLabeler* de Matlab, la cual permite generar regiones de interés que encierran puntos pertenecientes a la misma categoría mediante interpolación entre las nubes de puntos inicial y final anotadas manualmente (ver Fig. 3c). Finalmente, se lleva a cabo una normalización de las coordenadas entre 0 y 1 para hacer la CNN independiente de las dimensiones. Después de un proceso de entrenamiento con 700

nubes de puntos acondicionadas y 1160 iteraciones, se obtuvo una CNN que alcanzó un 99% de aciertos.

A. Localización del Puente

Utilizando los puntos etiquetados dentro de la categoría puente (no normalizados), se obtiene el promedio de la coordenada x de dichos puntos. Ahora, considerando que pueden existir algunos puntos mal clasificados por la CNN, se eliminan aquellos a una distancia mayor de 1 m respecto a la coordenada x promedio (x_{prom}) obtenida previamente y se recalcula nuevamente la coordenada x_{prom} con los puntos restantes. Este proceso se repite nuevamente eliminando ahora aquellos puntos a una distancia mayor de 0.5 m respecto a la coordenada x_{prom} calculada en la iteración anterior. El resultado obtenido se considera la localización del puente x_p en la dirección X_M .

B. Localización del Carro

Este elemento está acoplado al puente, así que su localización en dirección X_M coincide con la localización del puente. Para determinar su coordenada y , al igual que en el caso anterior, se seleccionan aquellos puntos etiquetados por la CNN dentro de la categoría carro cuya coordenada x sea cercana a la definida previamente para el puente (para evitar la inclusión de puntos mal clasificados por la CNN). Posteriormente, se obtiene la coordenada y promedio (y_{prom}) entre los puntos seleccionados. A continuación, se elimina un 20% de los puntos utilizados, específicamente aquellos cuya coordenada y está más distanciada respecto a y_{prom} . A partir de los puntos restantes se recalcula la coordenada y_{prom} , la cual es considerada como la localización del carro y_c en la dirección Y_M .

Un ejemplo de localización del puente y carro dentro de una nube de puntos 3D se muestra en la Fig. 4. Es importante señalar que, aunque el carro tiene mayores dimensiones (aproximadamente la de un prisma rectangular de $1 \times 0.5 \times 0.5$ m) que el gancho, la nube generada por el LiDAR 3D apenas posee puntos etiquetados como carro, lo cual dificulta su detección con la CNN. Por ello, la coordenada y_c del carro para una nube de puntos, se toma como elemento discriminatorio adicional en el análisis de la siguiente nube seleccionando aquellos puntos cuya coordenada y este cerca de la coordenada y_c anterior.

IV. DETECCIÓN DEL GANCHO Y DE LA CARGA MEDIANTE CLUSTERIZACIÓN

Una vez localizado el puente y el carro, se detectan y localizan el gancho y la carga (si está presente), utilizando un proceso de *clusterización* (agrupamiento) de puntos pertenecientes a estos elementos. El par de coordenadas (x_p, y_c) facilita la localización del gancho y la carga, ya que estos se encuentran ubicados en una posición similar, pero con una coordenada z inferior. Para ello, se utiliza una ventana cuadrada de búsqueda de dimensión τ (elegida por el usuario) y centrada en dichas coordenadas que es proyectada hacia abajo buscando los puntos dentro de la nube que estén localizados dentro de ella. Dichos puntos podrán pertenecer a tres elementos: cable, gancho y carga suspendida.

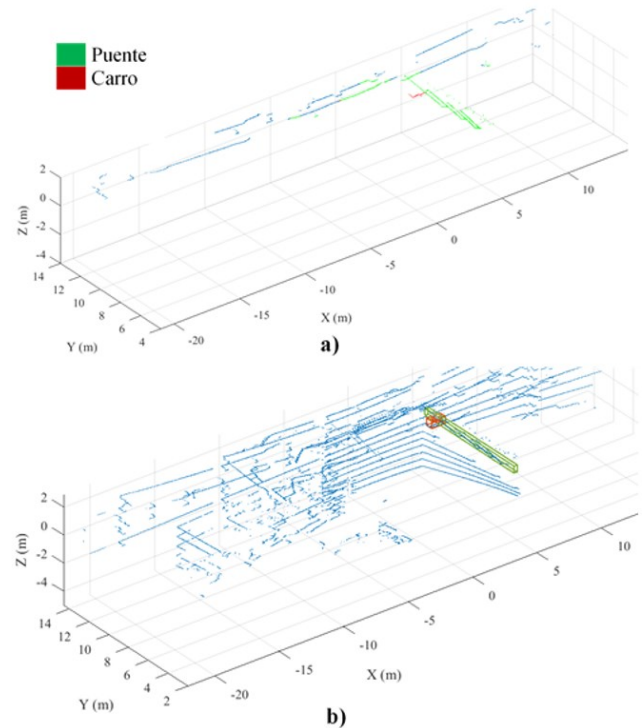


Fig. 4. Resultados de la localización del puente y del carro con la CNN (a) y sobreimpresión de ambos elementos en la nube de puntos 3D (b).

En la ventana de búsqueda también podrían seleccionarse puntos pertenecientes al suelo. Pero, en este caso, el campo de visión del LiDAR 3D no es suficiente para alcanzar esta superficie. Aunque, si se hubiese presentado esta condición, Matlab provee la función *segmentGroundFromLidarData* que permite retirar de una nube de puntos, aquellos que forman parte del suelo.

Una vez seleccionados los puntos dentro de la ventana de búsqueda, se realiza un proceso de *clusterización*, donde se agrupan puntos en un mismo conjunto con una distancia entre vecinos menor a 0.3 m. Posteriormente, se determinan las coordenadas medias ($\bar{x}_{ci}, \bar{y}_{ci}, \bar{z}_{ci}$) de cada conjunto c_i . Cuando el puente grúa no transporta carga (condición indicada por el usuario al sistema), se consideran las coordenadas medias del conjunto con cota más baja, como la localización del gancho (x_g, y_g, z_g). Un ejemplo de localización del gancho se muestra en la Fig. 5.

Este procedimiento puede fallar cuando el gancho pase sobre algún obstáculo. Por ello, se agregaron dos condiciones adicionales para la localización del gancho. Primero se considera que la ubicación inicial de este elemento corresponde a una zona segura (espacio donde no hay obstáculos cercanos), lo cual se garantiza al inicio de la monitorización del puente grúa. Segundo, una vez ubicado este elemento en una nube de puntos dada, se utiliza su coordenada z_g en la siguiente para descartar aquellos puntos cuya coordenada $z \leq (z_g - \Delta z_{max})$.

La elección de la constante Δz_{max} está asociada a la velocidad máxima de descenso del gancho que no permite que $\Delta z > \Delta z_{max}$ entre dos barridos láser consecutivos. Finalmente,

puede suceder que para una nube de puntos adquirida por el LiDAR 3D no se detecten puntos correspondientes al gancho. Esta situación es posible cuando el gancho esté lejos del sensor y quede ubicado entre dos rayos láser verticales del LiDAR 3D. En este caso, la posición del gancho se aproxima considerando su velocidad de avance calculada a partir de dos posiciones anteriores del mismo.

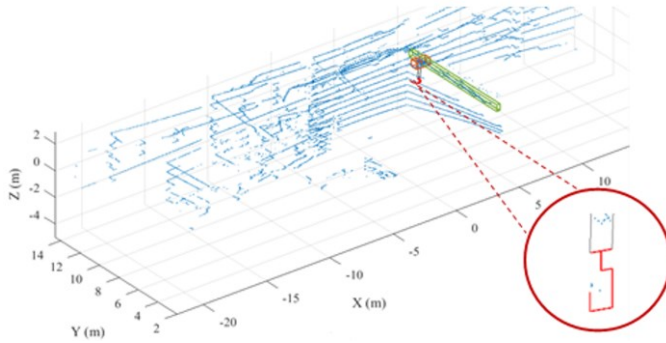


Fig. 5. Detalle de la sobreimpresión del gancho después de su búsqueda.

Por otra parte, cuando el usuario indica al sistema que el puente grúa está transportando carga, también se utiliza la ventana con centro en (x_p, y_c) y con dimensión τ para seleccionar los puntos dentro de ella. A continuación, se realiza la *clusterización* de puntos vecinos agrupándolos en conjuntos c_i (ver Fig. 6a) con coordenadas medias $(\bar{x}_{ci}, \bar{y}_{ci}, \bar{z}_{ci})$. Luego, para cada conjunto encontrado se determina la distancia d_{ci} entre su centroide y el eje Z_M tal que, $d_{ci} = \sqrt{\bar{x}_{ci}^2 + \bar{y}_{ci}^2}$. Esta distancia es relativamente constante en los conjuntos pertenecientes al cable y al gancho, pero existe una variación significativa en la carga, que es más ancha. Por ello, donde se establezca el mayor cambio de distancia entre dos conjuntos consecutivos se define la posición del gancho (x_g, y_g, z_g) (centroide del conjunto superior) y el primer conjunto de puntos perteneciente a la carga (centroide del conjunto inferior).

La carga queda conformada por todos los conjuntos de puntos ubicados bajo este primer conjunto (incluyéndolo) y está definida por un volumen cilíndrico que contiene todos los puntos pertenecientes a estos conjuntos (ver Fig. 6b). El eje de dicho cilindro corresponde a una recta paralela al eje Z_M que pasa por las coordenadas (x_g, y_g) del gancho. El radio de este cilindro r_i corresponde a la distancia desde el punto perteneciente a la carga que esté más alejado (x_p, y_p, z_p) del eje del cilindro, más una tolerancia τ_s definida para compensar la incertidumbre relacionada con zonas a la derecha e izquierda de la carga que no fueron escaneadas. Cuantitativamente, r_i está definido por:

$$r_i = \sqrt{(x_g - x_p)^2 + (y_g - y_p)^2} + \tau_s \quad (1)$$

Por otra parte, el centro del eje de este cilindro z_{cil} , está definido como la coordenada z promedio entre las coordenadas \bar{z}_{ci} pertenecientes a todos los conjuntos que conforman la carga. Así, la altura del cilindro es $h_i = z_{max} - z_{min} + 2\tau_h$, que toma en consideración la coordenada z del punto más bajo y alto

perteneciente a la carga z_{min} y z_{max} , respectivamente, junto con un excedente τ_h para tener en cuenta partes de la carga no detectadas, tanto por debajo como por encima de la misma.

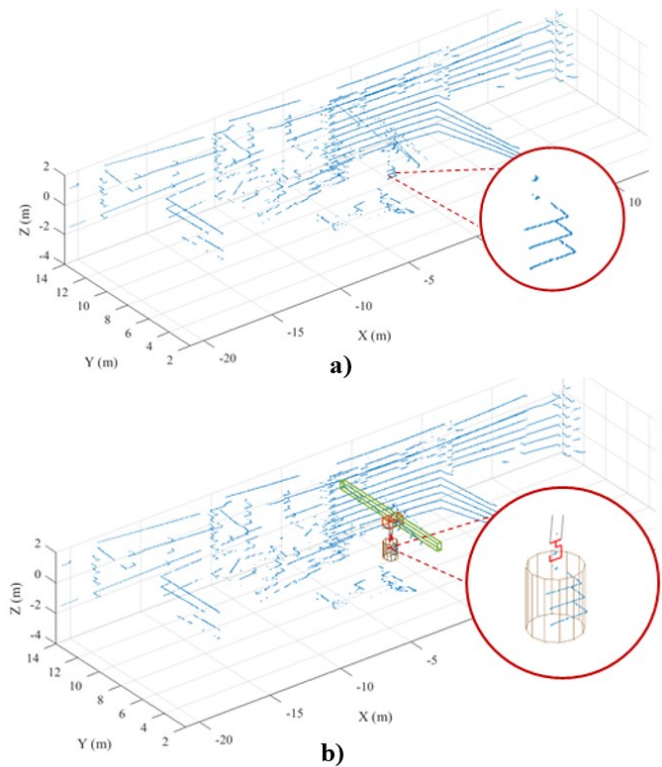


Fig. 6. Proceso de localización (a) y dimensionamiento de la carga (b).

Pueden ocurrir variaciones notables en las dimensiones de este volumen estimado a consecuencia de la aparición u omisión de rangos láser pertenecientes a la carga. Para evitar variaciones frecuentes, se tomó como radio del cilindro $r_{cil} = (3r_{i-1} + r_i)/4$ y altura $h_{cil} = (3h_{i-1} + h_i)/4$, que es la medida ponderada entre las magnitudes obtenidas para la nube de puntos actual (r_i, h_i) y las correspondientes para la nube de puntos anterior (r_{i-1}, h_{i-1}) siempre que $i > 1$.

No obstante, la exploración en la ventana de búsqueda puede seleccionar puntos de obstáculos como pertenecientes a la carga. Para evitar esto, también se consideraron las dos condiciones ya descritas para la detección del gancho: primero, considerar que la ubicación inicial de la carga corresponde a una zona segura (espacio donde no hay obstáculos cercanos), lo cual permite una localización inicial de la carga, en especial z_{cil} . Segundo, una vez ubicado este elemento, se utiliza su coordenada z_{cil} como condición discriminante en la selección de puntos cuando se realiza la búsqueda en el siguiente barrido: se descartan aquellos puntos cuya coordenada $z \leq z_{cil} - (h_{cil}/2 + \Delta z_{max})$.

V. DETECCIÓN DE COLISIONES

Se definieron dos métodos para la búsqueda de colisiones: uno omnidireccional y otro focalizado (ver Fig. 7). En ambos se establecieron tres volúmenes de interés. Primeramente, un volumen cilíndrico V_1 encierra la carga, cuyas dimensiones (h_{cil}, r_{cil}) se han estimado previamente. Cuando el puente grúa

no transporta carga y solo está presente el gancho, también se define un volumen cilíndrico V_1 cuyas dimensiones son suficientes para recubrir el gancho. En segundo lugar, un volumen V_2 que determina la posibilidad de colisión inminente, formado por un recubrimiento cilíndrico que rodea la carga, el cual posee un espesor τ_c . Por último, se presenta un tercer volumen V_3 mayor, cuya magnitud se define a partir de la distancia de seguridad d_s (ver Fig. 7).

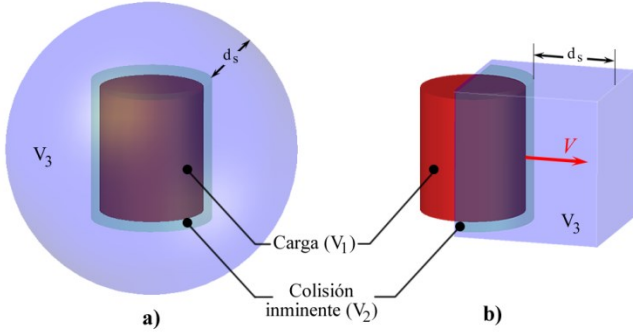


Fig. 7. Métodos para la detección de colisiones: omnidireccional (a) y focalizado (b).

La existencia de estos volúmenes permite definir tres niveles de riesgo: riesgo nulo R_0 , cuando todos los puntos de la nube no pertenecientes a la carga están fuera de los tres volúmenes. Luego, el riesgo R_1 , que indica la posibilidad de colisión cercana cuando un punto de la nube está dentro de V_3 y fuera de V_2 . Finalmente, el riesgo de colisión inminente R_2 , cuando un punto de la nube está dentro de V_2 .

A partir de estas definiciones se realiza el estudio de colisiones para cada nube de puntos escaneada por el LiDAR 3D, después de haberse producido la detección y dimensionamiento de la carga o gancho. En primer lugar, se descartan de la nube todos los puntos contenidos en V_1 puesto que estos pertenecen a la carga. Para el resto de puntos, se verifica su posición respecto a los volúmenes V_2 y V_3 . Si hay puntos de la nube ubicados dentro de estos espacios, se define el nivel de riesgo presente (R_1 ó R_2).

Por otra parte, la activación del riesgo R_2 no depende solamente de la presencia de un punto dentro de V_2 . Este riesgo se activa solo si en la nube previa se detectó un riesgo de colisión R_1 ó R_2 , lo cual indica que, efectivamente, la carga se estaba acercando a un obstáculo. Con esta limitante se intenta evitar la presencia de detecciones erróneas que se pueden producir cuando aparece un nuevo punto de la carga no escaneado previamente. Asimismo, cuando se detecta un riesgo R_2 , no se actualizan las dimensiones del cilindro que define la carga (h_{cil} , r_{cil}), sino que se considera $h_{cil} = h_{i-1}$ y $r_{cil} = r_{i-1}$. Con esta medida se intenta evitar que puntos pertenecientes a un obstáculo sean considerados como parte de la carga.

A. Búsqueda Omnidireccional

En este método, se produce la búsqueda de colisiones en todas las direcciones alrededor de la carga (ver Fig. 7a). Para ello se define el volumen V_3 como una esfera centrada en la carga, cuyo radio r_s vale:

$$r_s = \sqrt{(r_{cil} + \tau_c)^2 + (h_{cil}/2 + \tau_c)^2} + d_s \quad (2)$$

Cualquier punto de la nube no perteneciente a la carga que esté dentro de esta esfera y fuera de V_2 genera un riesgo de colisión R_1 . Para verificar si un punto $P = [p_x, p_y, p_z]$ está contenido en la esfera, se debe cumplir la condición:

$$\sqrt{(p_x - x_g)^2 + (p_y - y_g)^2 + (p_z - z_g)^2} \leq r_s \quad (3)$$

También se definieron tres volúmenes relacionados con el cable. Primero un cilindro que recubre el cable V_{1c} de radio r_{cab} . Segundo, un tubo cilíndrico con espesor τ_c llamada V_{2c} , la cual recubre a V_{1c} y define el riesgo de colisión inminente R_2 . Finalmente, otro tubo cilíndrico con espesor d_s llamada V_{3c} , la cual recubre a V_{2c} y define el riesgo de colisión cercana R_1 . La longitud de estos tres elementos se determina por la distancia desde la cara superior del cilindro que define la carga V_1 hasta el carro. En la Fig. 8a se muestra la esfera V_3 que recubre la carga y el tubo cilíndrico V_{3c} que recubre el cable. Cabe resaltar que este método es conservador, puesto que la dirección de desplazamiento del gancho o carga no apunta necesariamente hacia los puntos dentro de la esfera V_2 .

B. Búsqueda Focalizada

En este caso, la búsqueda se realiza solo en la dirección de desplazamiento de la carga o gancho. Así, el volumen de búsqueda V_3 queda definido como un prisma rectangular que parte desde la carga proyectándose hacia la dirección de avance definida por el vector de velocidad (ver Figs. 7b y 8b), el cual se obtiene a partir de la posición actual de la carga (x_g, y_g, z_g) , la posición anterior $(x_{g\ i-1}, y_{g\ i-1}, z_{g\ i-1})$, y el intervalo de tiempo Δt entre los dos barridos láser consecutivos. Este vector de velocidad $\vec{V} = [v_x, v_y, v_z]$ queda determinado por:

$$\vec{V} = \left[\left(\frac{x_g - x_{g\ i-1}}{\Delta t} \right), \left(\frac{y_g - y_{g\ i-1}}{\Delta t} \right), \left(\frac{z_g - z_{g\ i-1}}{\Delta t} \right) \right] \quad (4)$$

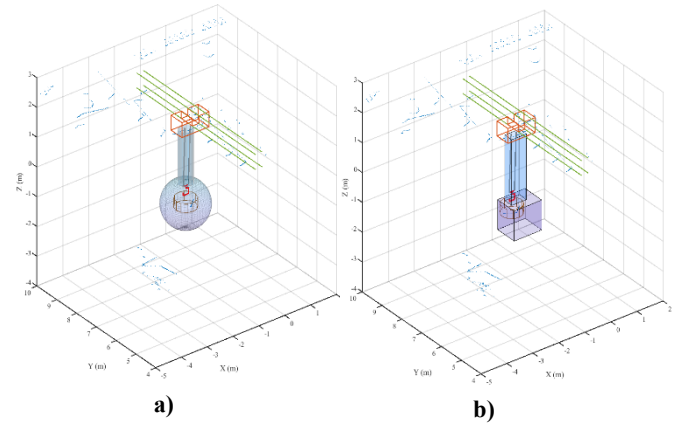


Fig. 8. Sobreimpresión de los volúmenes V_3 y V_{3c} sobre una nube de puntos 3D para la detección de colisión omnidireccional (a) y focalizada (b).

Para la construcción del prisma se define, en primer lugar, un nuevo sistema de referencia $X_V Y_V Z_V$ cuyo origen es el centro de la carga (x_g, y_g, z_g) o del gancho, si no hay carga acoplada. La dirección del eje Z_V coincide con la dirección de desplazamiento dada por \vec{V} y está dada por el vector unitario

$U_z = \vec{V}/|\vec{V}|$ (ver Fig. 9a). El eje X_V pertenece a un plano formado por \vec{V} y una recta vertical que pasa por el centro de la carga. Para definir la dirección de X_V se obtiene primero \vec{X}_V a través de:

$$\begin{cases} \vec{X}_V = \left[v_x, v_y, -\frac{(v_x^2 + v_y^2)}{v_z} \right] & \text{si } v_z \neq 0 \\ \vec{X}_V = [0,0,1] & \text{si } v_z = 0 \end{cases} \quad (5)$$

después, la dirección a través del vector unitario $U_x = \vec{X}_V/|\vec{X}_V|$. Finalmente, el eje Y_V se define para completar el sistema dextrógiro: $U_y = U_z \times U_x$.

La base rectangular del prisma V_3 está contenida en el plano $X_V Y_V$ y centrada en el origen de este sistema de referencia. El ancho de esta cara es $a_p = 2(r_{cil} + \tau_c)$ y su altura $h_p = h_{cil} + 2\tau_c$. Dicha cara se proyecta en la dirección Z_V para formar el prisma 3D con una distancia $d_p = r_{cil} + \tau_c + d_s$ (ver Fig. 9a). Para evaluar si los puntos no pertenecientes a la carga están contenidos en el prisma de búsqueda, se transforman las coordenadas de los puntos desde el sistema $X_M Y_M Z_M$ al sistema $X_V Y_V Z_V$ a través de:

$${}^V P = [{}^V x_p, {}^V y_p, {}^V z_p, 1]^T = {}^V T_M {}^M P \quad (6)$$

donde ${}^M P$ corresponde a las coordenadas homogéneas de un punto de la nube en el sistema $X_M Y_M Z_M$, ${}^V P$ contiene las coordenadas equivalentes de dicho punto en el sistema $X_V Y_V Z_V$ y ${}^V T_M$ es la matriz de transformación homogénea entre estos dos sistemas definida por:

$${}^V T_M = \begin{bmatrix} U_x & -U_x O_V^T \\ U_y & -U_y O_V^T \\ U_z & -U_z O_V^T \\ 0 & 1 \end{bmatrix} \quad (7)$$

siendo $O_V = [x_g, y_g, z_g]$ las coordenadas del origen del sistema $X_V Y_V Z_V$ medidas en el sistema $X_M Y_M Z_M$. Una vez transformadas las coordenadas de los puntos al sistema $X_V Y_V Z_V$, se utiliza:

$${}^V P \in V_3 \text{ si } \begin{cases} -\frac{h_p}{2} \leq {}^V x_p \leq \frac{h_p}{2} \\ \quad \quad \quad \wedge \\ -\frac{a_p}{2} \leq {}^V y_p \leq \frac{a_p}{2} \\ \quad \quad \quad \wedge \\ 0 \leq {}^V z_p \leq d_p \end{cases} \quad (8)$$

para verificar si están contenidos en V_3 , lo cual indica un riesgo de colisión cercana R_1 . Posteriormente también se debe verificar si el mismo punto está contenido en V_2 para verificar el riesgo de colisión inminente R_2 .

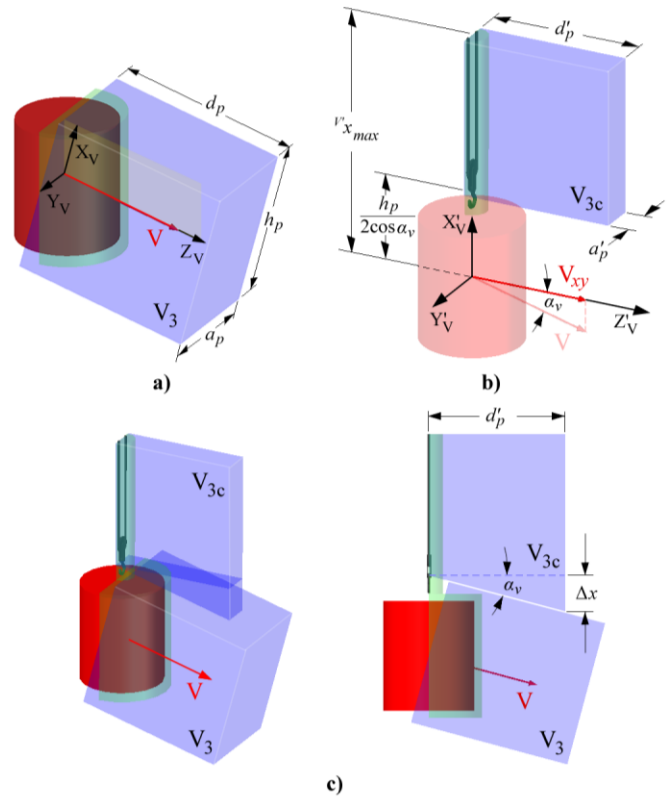


Fig. 9. Volúmenes para la búsqueda de colisiones focalizada de la carga (a), de los cables (b) y ajuste del espacio entre ambas (c).

En el caso del cable, también se genera una búsqueda focalizada a través de otro prisma rectangular V_{3c} que se proyecta en la dirección de avance horizontal del cable dada por $\vec{v}_{xy} = [v_x, v_y, 0]$. Para ello, se construye un nuevo sistema de referencia $X'_V Y'_V Z'_V$ con origen en el centro de la carga, dirección del eje Z'_V coincidente con la dirección de avance horizontal del cable dada por el vector unitario $U'_z = \vec{v}_{xy}/|\vec{v}_{xy}|$ (ver Fig. 9b). La dirección de X'_V es vertical dada por $U'_x = [0,0,1]$ y la dirección de Y'_V se obtiene por medio de $U'_y = U'_z \times U'_x$ para completar el sistema dextrógiro.

A continuación se define el prisma V_{3c} (ver Fig. 9b). Se consideró que su base rectangular está contenida en el plano $X'_V Y'_V$, el ancho de esta cara es $a'_p = 2(r_{cab} + \tau_c)$ y su altura está definida por la arista inferior, la cual es horizontal y ubicada en la coordenada ${}^V x = h_p/(2 \cos \alpha_v)$ donde, $\alpha_v = \cos^{-1}(|\vec{v}_{xy}|/|\vec{V}|)$, mientras que su arista superior alcanza la coordenada ${}^V x_{max} = (0.8 - z_g)m$. Este prisma se proyecta hacia la dirección de avance horizontal del cable con una longitud $d'_p = \min(r_{cab} + \tau_c + d_s, d_p \cos \alpha_v)$, aunque durante la proyección, la altura de este prisma varía hasta alcanzar un incremento $\Delta x = -d'_p \tan \alpha_v$ (ver Fig. 9c), el cual permite que no quede espacio entre éste y el prisma de búsqueda de la carga.

Para verificar si un punto no perteneciente a la carga está contenido en el prisma V_{3c} , se transforman las coordenadas de dicho punto desde el sistema $X_M Y_M Z_M$ al $X'_V Y'_V Z'_V$ aplicando:

$${}^V P = [{}^V x_p, {}^V y_p, {}^V z_p, 1]^T = {}^V T_M M P \quad (9)$$

donde ${}^V T_M$ es la matriz de transformación homogénea entre estos sistemas, definida a través de:

$${}^V T_M = \begin{bmatrix} U'_x & -U'_x O_V^T \\ U'_y & -U'_y O_V^T \\ U'_z & -U'_z O_V^T \\ 0 & 1 \end{bmatrix} \quad (10)$$

Una vez transformadas las coordenadas de los puntos al sistema $X'_V Y'_V Z'_V$ se puede analizar el riesgo verificando la presencia de cada punto dentro de los volúmenes V_{2c} y V_{3c} . Por ejemplo, para evaluar la posibilidad de colisión cercana R_1 de un punto: ${}^V P \in V_{3c}$, basta verificar que:

$$\begin{cases} \frac{h_p}{2 \cos \alpha_v} - {}^V z_p \tan \alpha_v \leq {}^V x_p \leq {}^V x_{max} \\ \wedge \\ -\frac{a'_p}{2} \leq {}^V y_p \leq \frac{a'_p}{2} \\ \wedge \\ 0 \leq {}^V z_p \leq d'_p \end{cases} \quad (11)$$

Si un punto está dentro del volumen V_{2c} , la condición de riesgo aumenta a colisión inminente R_2 .

VI. RESULTADOS EXPERIMENTALES

El sistema de detección propuesto ha sido programado en una aplicación desarrollada en Matlab (ver Fig. 10), donde el usuario puede, en primer lugar, configurar el sistema, seleccionando entre gancho vacío o con carga, el método para la búsqueda de colisiones (omnidireccional o focalizada) y las opciones de visualización deseadas: ver nube de puntos, representación del gancho y la carga, volúmenes de búsqueda de colisiones junto con los puntos resaltados que pueden ocasionar colisiones, y trazar la trayectoria seguida por la carga al final de la monitorización.

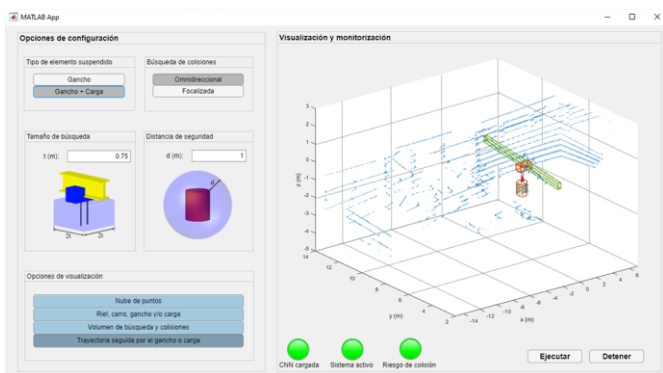


Fig. 10. Aplicación desarrollada en MATLAB para la monitorización del puente grúa.

El usuario también puede especificar el tamaño de la ventana de búsqueda τ para la localización del gancho y de la carga, y la distancia de seguridad d_s necesaria para la búsqueda de colisiones. Al ejecutar la aplicación, esta adquiere nubes de puntos, realiza todo el proceso y genera, a través de un conjunto

de indicadores, las advertencias respecto al nivel de riesgo detectado (verde para R_0 , naranja para R_1 y rojo para R_2) mientras va mostrando una representación gráfica del proceso de acuerdo con las opciones seleccionadas por el usuario (ver Fig. 11).

Se hicieron múltiples pruebas de la aplicación en una computadora equipada con un procesador Intel i5-4460 de cuatro núcleos a 3.2 GHz y 8 GB de RAM. Primero, hubo tres pruebas para la detección solo del gancho, con 200 nubes de puntos 3D cada una y un porcentaje de acierto del 91.66% en promedio. Destaca la tercera prueba, por su menor porcentaje de aciertos, al partir el puente grúa de una posición desfavorable para la detección del carro. Por ello, en las 34 primeras nubes, correspondientes a los 6.8 s iniciales, no hubo identificación del carro. Pero, una vez que la CNN detectó el carro, se localizó correctamente el gancho.

Adicionalmente, se hicieron seis pruebas análogas con 600 nubes en cada una, considerando el manejo de carga. En este caso, el porcentaje de acierto se incrementó a 99.96%. Esta mejoría se debe principalmente a que las dimensiones de la carga son superiores a las del gancho, lo cual permite obtener un mayor número de puntos de la carga que facilitan su detección. La mitad de pruebas se realizó con una caja de dimensiones 0.42x0.58x0.28 m, mientras que las restantes se llevaron a cabo con una caja mayor de 0.64x0.64x0.41 m.

Por otra parte, se estudió el tiempo que tarda la computadora en ejecutar todas las tareas que han sido programadas para cada barrido láser. En las pruebas realizadas se encontró que la detección del carro y del puente utilizando la CNN tarda aproximadamente 0.3 s, mientras que la detección de puente, carro y gancho sumada con la búsqueda de colisiones (sin animación) promedia 0.345 s. Si se incluye el proceso de representación gráfica con animación, el tiempo promedio se incrementa a 0.427 s. En el caso de la detección de carga (más puente y carro) y búsqueda de colisiones, el tiempo promedio se incrementa a 0.38 s sin animación. Si se incluye la representación gráfica, el tiempo promedio total se sitúa alrededor de 0.486 s.

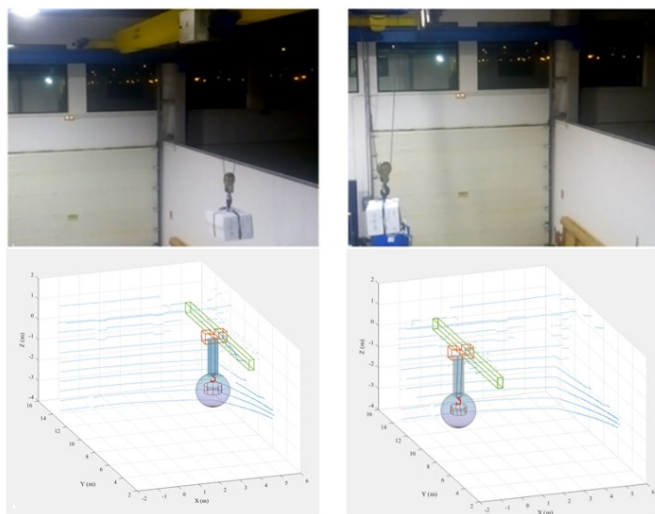


Fig. 11. Ejemplos de monitorización de la carga durante su traslado.

Considerando que el LiDAR 3D realiza un barrido del almacén cada 0.2 s, esto implica que la aplicación no puede procesar todos los barridos adquiridos por el sensor, pero sí puede analizar uno de cada dos si se omite la representación gráfica de las nubes en la pantalla.

VII. CONCLUSIONES

Este trabajo ha permitido verificar favorablemente la capacidad que tiene un LiDAR 3D para la monitorización de puentes grúa. En comparación con la visión artificial, tiene a su favor ser independiente de las condiciones de iluminación del espacio de trabajo y de tener un menor número de datos (rangos) que procesar que las imágenes (píxeles).

Se ha desarrollado una aplicación en Matlab que es capaz de monitorizar un puente grúa, detectando la posición de dos de sus elementos (puente y carro) con una CNN y de los otros dos (gancho y carga) mediante agrupamiento de puntos láser. Además, permite estimar la posibilidad de colisión de acuerdo con tres niveles de riesgo y con dos métodos diferentes de detección.

La capacidad del sistema desarrollado puede incrementarse con una computadora más potente y con una mayor densidad de puntos, lo cual se puede lograr con un LiDAR 3D de mayor resolución vertical. A partir de este trabajo, se pueden proponer otras áreas de investigación que incluyen: el dimensionamiento y estimación del volumen de la carga, estudio de colisiones con obstáculos móviles, planificación de trayectorias seguras e integración de este sistema con otros para lograr la automatización completa de los puentes grúa.

AGRADECIMIENTO

J. M. García agradece a la Fundación Carolina de España por la beca otorgada para la realización de este trabajo.

REFERENCIAS

- [1] O. Lengerke, J. Niño, M. Suell, and M. Morales, "Diseño y automatización de un puente grúa para procesos secuenciales," in *8vo Congreso Iberoamericano de Ingeniería Mecánica*, Cusco, 2007, pp. 1-8.
- [2] A. Akhtulov, M. Kirasirov, and O. Kirasirov, "The construction of the system of automation of designing of load-lifting cranes of bridge type," *Journal of Physics: Conference Series*, vol. 1399, no. 4, pp. 1-5, 2019. DOI:10.1088/1742-6596/1399/4/044031.
- [3] J. López, "Actualización del sistema de automatización, control y supervisión de una grúa pórtico," Universidad de Oviedo, España, Trabajo fin de máster 2018.
- [4] J. Carcedo, "Ingeniería de automatización, control y supervisión de una grúa semi-pórtico," Universidad de Oviedo, España, Trabajo Fin de Máster 2020.
- [5] S. Beller and H. Yavuz, "Crane automation and mechanical damping methods," *Alexandria Engineering Journal*, vol. 60, pp. 3275-3293, 2021. DOI:10.1016/j.aej.2021.01.048.
- [6] H. Bao, J. An, M. Zhou, and Q. Kang, "A Data-driven MPC Algorithm for Bridge Cranes," in *International Conference on Advanced Mechatronic Systems*, Hanoi, 2020, pp. 328-332. DOI:10.1109/ICAMEchS49982.2020.9310150.
- [7] P. Hyla, "The crane control systems: A survey," in *17th International Conference on Methods & Models in Automation & Robotics (MMAR)*, Miedzydroje, 2012, pp. 505-509. DOI:10.1109/MMAR.2012.6347867.
- [8] P. Hyla and J. Szpytko, "The vision technique concept support crane safety exploitation process," *Journal of KONBiN*, vol. 49, no. 9, pp. 37-47, 2019. DOI:10.2478/jok-2019-0074.
- [9] J. Huang, W. Xu, W. Zhao, and H. Yuan, "An improved method for swing measurement based on monocular vision to the payload of overhead crane," *Transactions of the Institute of Measurement and Control*, vol. 44, no. 1, pp. 1-10, 2020. DOI:10.1177/0142331220921318.
- [10] N. Sun, J. Zhang, T. Yang, Y. Fu, and M. Zhou, "A Flexible Rope Crane Control Method," in *IEEE International Conference on Real-time Computing and Robotics*, Kandima, 2018, pp. 254-258. DOI:10.1109/RCAR.2018.8621682.
- [11] A. Kaneshige, T. Akamatsu, and K. Terashima, "The development of an autonomous overhead traveling crane with real time path planning based on the potential method," in *International Conference on Control and Automation (ICCA)*, Budapest, 2005, pp. 1079-1084. DOI:10.1109/ICCA.2005.1528282.
- [12] K. Zhao, Q. Zhou, X. Xiong, and J. Zhao, "The construction method of the digital operation environment for bridge cranes," *Mathematical problems in engineering*, pp. 1-13, 2021. DOI: 10.1155/2021/5528639.
- [13] C. Zhengyan, Z. Zhengwei, D. Qiang, and L. Zheren, "Route planning of intelligent bridge cranes based on an improved artificial potential field method," *Journal of Intelligent & Fuzzy Systems*, vol. Pre-press., pp. 1-8, 2021. DOI:10.3233/JIFS-189696.
- [14] C. Zhimei, L. Min, S. Xuejuan, and Z. Zhicheng, "Obstacle avoidance path planning of bridge crane based on improved RRT algorithm," *Journal of system simulation*, vol. 33, no. 8, pp. 1832-1838, 2021. DOI: 10.16182/j.issn1004731x.joss.20-0272.
- [15] J. Yang, M. Huang, W. Chien, and M. Tsai, "Application of machine vision to collision avoidance control of the overhead crane," in *International Conference of Electrical, Automation and Mechanical Engineering*, 2015, pp. 361-364. DOI:10.2991/eamc-15.2015.100.
- [16] A. Kaneshige, S. Nagai, S. Ueki, T. Miyoshi, and K. Terashima, "An algorithm for on-line path planning of autonomous mobile overhead traveling crane based on obstacle information," in *IEEE International Conference on Automation Science and Engineering (CASE)*, Seoul, 2012, pp. 914-919. DOI:10.1109/CoASE.2012.6386356.
- [17] S. Nagai, A. Kaneshige, and S. Ueki, "Three-dimensional obstacle avoidance online path-planning method for autonomous mobile overhead crane," in *2011 IEEE International Conference on Mechatronics and Automation*, Beijing, 2011, pp. 1497-1502. DOI:10.1109/ICMA.2011.5985971.
- [18] H. Chen, G. Liu, G. Tian, J. Zhang, and Z. Ji, "Adaptive Safe Distance Prediction Using MPC for Bridge Cranes Considering Anti-Swing," in *2020 Chinese Automation Congress (CAC)*, Shanghai, 2020, pp. 1914-1919. DOI:10.1109/CAC51589.2020.9326855.
- [19] Y. Zhou, H. Guo, L. Ma, Z. Zhang, and M. Skitmore, "Image-based onsite object recognition for automatic crane lifting tasks," *Automation in construction*, vol. 123, pp. 1-13, 2021. DOI:10.1016/j.autcon.2020.103527.
- [20] Q. Xie, H. Wen, and Q. Wang, "Research and Reliability Analysis of Double Bridge Crane Control System," *International Journal of Mechatronics and Applied Mechanics*, no. 4, pp. 188-195, 2018.
- [21] T. Raj, F. H. Hashim, A. B. Huddin, M. F. Ibrahim, and A. Hussain, "A survey on LiDAR scanning mechanisms," *Electronics*, vol. 9, no. 5, pp. 1-25, 2020. DOI: 10.3390/electronics9050741.
- [22] W. J. Zhang, G. Yang, Y. Lin, C. Ji, and M. M. Gupta, "On Definition of Deep Learning," in *World Automation Congress*, Stevenson, 2018, pp. 1-5. DOI: 10.23919/WAC.2018.8430387.
- [23] S. Cohen, "The basics of machine learning: strategies and techniques," in *Artificial Intelligence and Deep Learning in Pathology*: Elsevier, 2021, pp. 13-40. DOI:10.1016/B978-0-323-67538-3.00002-6.
- [24] L. Soto-Ayala and J. Cantoral-Ceballos, "Automatic blood-cell classification via convolutional neural networks and transfer learning," *IEEE Latin America Transactions*, vol. 19, no. 12, 2021. DOI: 10.1109/TLA.2021.9480144.
- [25] A. Asvadi, L. Garrote, C. Premebida, P. Peixoto, and U. Nunes, "DepthCN: vehicle detection using 3D-LIDAR and ConvNet," in *IEEE*

- 20th International Conference on Intelligent Transportation Systems*, Yokohama, 2017, pp. 1-6. DOI: 10.1109/ITSC.2017.8317880.
- [26] P. Wei, L. Cagle, T. Reza, J. Bal, and J. Gafford, "LiDAR and camera detection fusion in a real-time industrial multi-sensor collision avoidance system," *Electronics*, vol. 7, no. 0, pp. 1-34, 2018. DOI:10.3390/electronics7000000.
- [27] S. Mihai, P. Shah, G. Mapp, H. Nguyen, and R. Trestian, "Towards autonomous driving: A machine learning-based pedestrian detection system using 16-layer LiDAR," in *13th International Conference on Communications*, Bucharest, 2020, pp. 271-276. DOI: 10.1109/COMM48946.2020.9142042.
- [28] J. Szpytko and P. Hyla, "Material handling devices operation environment 3D-type presentation based on laser scanning systems," *Journal of KONES Powertrain and Transport*, vol. 17, no. 2, pp. 451-458, 2010.
- [29] H. Jeong, H. Hong, G. Park, M. Won, M. Kim, and H. Yu, "Point cloud segmentation of crane Parts using dynamic graph CNN for crane collision avoidance," *Journal of Computing Science and Engineering*, vol. 13, no. 3, pp. 99-106, 2019. DOI: 10.5626/JCSE.2019.13.3.99.
- [30] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "PointNet++: Deep hierarchical feature learning on point sets in a metric space," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, Long Beach, 2017, pp. 5105-5114. DOI:10.5555/3295222.3295263.
- [31] J. Qiao, Y. Lv, C. Cao, Z. Wang, and A. Li, "Multivariate deep learning classification of Alzheimer's disease based on hierarchical partner matching independent component analysis," *Frontiers in Aging Neuroscience*, vol. 10, pp. 1-12, 2018. DOI:10.3389/fnagi.2018.00417.



Jesús M. García received his PhD degree from the Universidad de Málaga, Spain (2015), and Mechanical Engineer degree from Universidad Nacional Experimental del Táchira (UNET), Venezuela (2001). His main research interest focuses on mechanical design and mobile robotics. He is Titular Professor at UNET and head of its Laboratorio de Prototipos.



Jorge L. Martínez (M'11-SM'21) received the M.Eng. and Ph.D. degrees in Computer Science from the University of Málaga (UMA), Málaga, Spain, in 1991 and 1994, respectively. From 2017, he is Full Professor at the Department of Systems Engineering and Automation of UMA. His teaching

experience include several process control and robotics lectures and the supervision of 5 Ph.D. theses. His 28 research papers indexed in the Journal Citation Reports are about different aspects of mobile robotics.



Antonio J. Reina has a degree in Computer Science (1991) and a PhD in Computer Engineering (2001) from the University of Málaga (UMA). His research activity has been developed since 1993 in the Department of Systems Engineering and Automation, where he is currently Associate Professor (2003).

His research production includes more than 30 publications, of which 8 are journal papers indexed in the Journal Citation Reports, including a contribution about astronomy to Nature.