Urban Dual Mode Video Detection System Based on Fisheye and PTZ Cameras

Sebastián Arroyo, Lilian García, Félix Safar, Damián Oliva

Abstract— **This work presents an artificial vision-based monitoring system for urban environments. It comprises a fisheye camera monitoring the scene's 180ºx360º hemisphere and a Pan-Tilt-Zoom camera capturing narrower regions of interest in high-resolution. The ONVIF protocol standard is used to interface both IP-cameras, allowing for the integration of camera control (image acquisition and movement) and geometric calculations on a single device. The events of interest (motion of vehicles and pedestrians) are assumed to happen on the ground plane. This assumption is required to solve the back-projection, the function that maps coordinates in the highly distorted images of the fisheye camera to the ground plane. A calibration strategy estimates the poses of the cameras without placing restrictions on their orientations or relative distance. It optimizes the back-projection error in the ground plane instead of the re-projection error in the image. Finally, a simple pointing and zoom adjustment strategy controls the Pan-Tilt-Zoom camera. The system is tested in controlled laboratory conditions and shows accurate outdoor performance for pedestrian observation.**

Index Terms— **Fisheye, omnidirectional camera, PTZ, ONVIF, camera calibration.**

I. Introduccion

El bajo costo de las cámaras de tecnología IP y el crecimiento de estas redes son una opción viable monitoreo visual de escenas urbanas [1]. Esto permite la integración de información proveniente de distintas cámaras mejorando la estimación del estado de los objetos en la escena. Las cámaras tradicionales (fijas o Pan-Tilt-Zoom, PTZ) tienen un campo de visión máximo de 60° × 60°. Las cámaras omnidireccionales con visión de campo amplio (VCA) (por ejemplo, las cámaras *fisheye*, FE) permiten medir un hemisferio completo de la escena (360° × 180°) [2]-[4]. Este aumento en el campo visual es una ventaja significativa en la reducción de puntos ciegos del sistema. Sin embargo, el aumento del campo visual produce la aparición de distorsiones y una reducción en la resolución de la imagen.

Para conjugar los beneficios de ambas cámaras (FE y PTZ), existen desarrollos previos de sistemas de visión compuestos por cámaras tradicionales y cámaras omnidireccionales catadióptricas o dióptricas [7]-[8], [21]-[22], (Fig. 1A). Estos sistemas se basan en técnicas de reconstrucción 3D. Primero se corrigen las distorsiones en la cámara VCA y se genera una cámara de perspectiva (virtual). Luego se asocian los puntos 2D en ambas cámaras que representan el mismo punto 3D de la escena. Finalmente, se utilizan técnicas tradicionales de triangulación para posicionar el punto de interés en la escena.

En los sistemas de monitoreo urbano, los eventos de mayor interés se producen sobre la superficie terrestre, estando asociados al comportamiento de vehículos y/o peatones [2], [4]-[5]. Si bien esta suposición introduce la restricción asociada al movimiento sobre la superficie terrestre, permite el posicionamiento de la cámara PTZ utilizando únicamente la detección del objeto en la cámara FE. Este enfoque desarrollado en [6], simplifica notablemente la estrategia de apunte de la cámara PTZ, ya que no es necesario aplicar algoritmos de coincidencia estéreo (por ejemplo a través de técnicas de *feature-matching*) que aseguren que el punto de interés observado en ambas cámaras es el mismo y de este modo, poder realizar la triangulación antes mencionada.

Sin embargo, la solución presentada en [6] asume que las bases de las cámaras VCA y PTZ son perpendiculares a la superficie de observación y que su posición relativa es conocida. Esta suposición pierde validez cuando las cámaras FE o PTZ se instalan con una inclinación respecto a la normal a la superficie. Esto puede deberse a errores de instalación o a que se desea capturar alguna dirección de la escena con mayor resolución. También pierde validez cuando las cámaras son instaladas a distintos tiempos y en posiciones alejadas, desconociendo a priori sus coordenadas relativas.

Los aportes principales de este trabajo son: a) desarrollar una metodología de calibración para el caso general de poses arbitrarias en ambas cámaras; b) calcular la función de retro-proyección desde la imagen FE al plano de observación; y c) proponer una estrategia simple para el apunte y ajuste de zoom en la cámara PTZ.

Para la calibración, se parte de un método tradicional que estima la pose de las cámaras minimizando el error de proyección sobre la imagen. Luego, se muestra que es necesario optimizar el error de retro-proyección sobre el plano de observación para mejorar sustancialmente la predicción de la posición del objeto y así lograr un apunte satisfactorio de la cámara PTZ. La solución propuesta no necesita la corrección *online* de distorsiones en la imagen de la cámara VCA, ni tampoco la aplicación algoritmos de coincidencia estéreo para lograr el apunte de la cámara PTZ.

La organización del trabajo es la siguiente: En la Sección II se describen las cámaras y el protocolo de comunicación utilizado. En la Sección III se detallan los modelos utilizados y el proceso de calibración intrínseca. En la Sección IV se describe el proceso de estimación de pose de las cámaras (calibración extrínseca) y el método de predicción de la posición sobre el plano de observación. En la Sección V, se

Este trabajo fue financiado por la Universidad Nacional de Quilmes, Programa I+D UNQ1303/19. Sebastián Arroyo, Felix Safar y Damián Oliva están en la Universidad Nacional de Quilmes, (e-mail: doliva@unq.edu.ar).

describe una estrategia para el apunte de la cámara PTZ y un método simple para el ajuste del *zoom*. Finalmente, se testea experimentalmente todo el sistema para situaciones reales en ambientes interiores y exteriores.

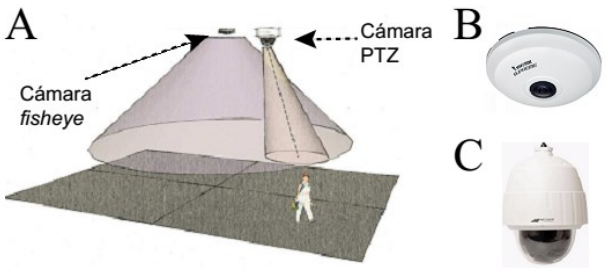

Fig. 1. (A) Sistema propuesto con visión de campo amplio (cámara *fisheye*, FE) y de alta resolución (cámara PTZ). (B) Cámara fisheye VIVOTEK FE8172 [20]. (C) Cámara PTZ Hecker HE-Z9116.

II. Cámaras Utilizadas e Implementación del Protocolo ONVIF

A. Cámaras Utilizadas

La cámara *fisheye* (FE) que se utilizó (también denominada VCA) es de la marca VIVOTEK modelo FE8172 (Fig. 1B), con las siguientes características: 1) Lente de tipo ojo de pez con campo visual de 360°×183°. 2) Tasa máxima de adquisición de fotogramas por segundo de 12 fps con una resolución máxima de 1920 × 1920 píxeles.

La cámara PTZ utilizada es una cámara domo de la marca Hecker modelo HE-Z9116 (Fig. 1C), con las siguientes características: 1) Rango de alcance de 360º en el plano horizontal, 180º en el vertical y *zoom* óptico de 30×. 2) Tasa máxima de adquisición de datos de 25 fps con una resolución de 1280 × 960 píxeles. 3) Modo de enfoque automático, semiautomático y manual.

B. Protocolo ONVIF

ONVIF, siglas que responden a *Open Network Video Interface Forum*, es un protocolo abierto que proporciona una única interfaz para dispositivos de seguridad basados en IP, independientemente de la marca del fabricante. El protocolo contiene un conjunto de especificaciones basadas en los estándares de *Web Services* y se divide en dos grupos: uno dedicado al ***core***, implementado en todos los dispositivos compatibles con ONVIF y otro dedicado a ***services***, que depende del perfil del equipo. Las funcionalidades de ambos grupos se detallan en el Tabla I.

Los servicios web que contienen las funciones del protocolo se encuentran generados en documentos WSDL (*Web Service Description Language*).

Las operaciones que se usaron en el desarrollo corresponden a especificaciones del *core* [10], de video, audio, de movimiento en los ejes y ajuste del *zoom*. La comunicación con el servidor se realiza con SOAP (*Simple Object Access Protocol*), que permite el intercambio entre aplicaciones que corren en cualquier sistema operativo y que pueden estar implementadas en cualquier lenguaje. SOAP es un protocolo de mensajería basado en XML que define la estructura con la que se arman los mensajes y provee una convención para hacer llamadas a procedimientos y sus respuestas. La interacción con las operaciones de ONVIF se realiza a través de estos mensajes que contienen los parámetros de entrada y devuelven otros con el resultado de su ejecución [11].

TABLA I
FUNCIONALIDADES DEL PROTOCOLO ONVIF

Especificaciones del Core	Especificaciones de servicios
Configuración IP del equipo	Configuración de los perfiles media
Detección de dispositivos en red	Control de accesos y permisos
Administración de dispositivos	Entrada/Salida de dispositivos
Gestión de eventos y alarmas	Control de movimiento PTZ
Visualización en tiempo real	Control de grabaciones
Seguridad	Análisis de video
	Seguridad avanzada

La transmisión de la imagen y el audio se realiza utilizando el estándar RTP/RTSP y también se apoya en estándares de compresión de video como H.264, MPEG-4 y M-JPEG y audio como G.711, G.726, AAC y unidireccional.

Para acceder a las funcionalidades de los equipos mediante las operaciones del protocolo ONVIF se utiliza un cliente denominado *python-onvif-zeep*. El cliente está codificado en Python por Quatanium Co [12].

III. Modelos de Cámaras y Calibración Intrínseca

A. Calibración Intrínseca de la Cámara PTZ

La cámara PTZ se enmarca en el modelo *pinhole* que describe la relación matemática entre las coordenadas de un punto 3D en el mundo ${}^{M}p=(X,Y,Z)^{T}$ y su proyección en el plano de la imagen *(u, v),* (ver Fig. 2A)[13].

Los parámetros extrínsecos de la cámara PTZ (que denominaremos $\gamma_{e,PTZ}$) corresponden al vector de traslación ${}^{C}t_{M}$ del origen de la trama mundo {M} respecto a la trama cámara {C}; y a la matriz de rotación ${}^{C}_{M}R$, que describe la orientación de la trama {M} con respecto a la trama {C} (Fig. 2A) [14]-[15]. Las coordenadas de un punto *p* respecto a la trama {C}, ${}^{C}p=(x,y,z)^{T}$, están relacionadas con las coordenadas respecto al mundo ${}^{M}p=(X,Y,Z)^{T}$, a través de (1):

$$ {}^{C}p = {}^{C}_{M}R \cdot {}^{M}p + {}^{C}t_{M} \tag{1} $$

Las coordenadas de la proyección del punto *p* sobre el CCD de la cámara PTZ están dadas por *(u, v)* y se calculan con la transformación (2):

$$ \begin{aligned} x' &= x/z \\ y' &= y/z \\ u &= f_x \cdot x' + c_x \\ v &= f_y \cdot y' + c_y \end{aligned} \tag{2} $$

Donde (f_x, f_y) corresponde a las distancias focales y (c_x, c_y) es el punto principal. Estos son los parámetros intrínsecos del modelo de la cámara PTZ (que denominaremos $\gamma_{i,PTZ}$) que se estiman a partir de un proceso de calibración utilizando funciones de la librería OpenCV [13] que están basadas en el trabajo [24]. Notar que las variables *(x', y')* corresponden a las

coordenadas homogéneas y solo son utilizadas en cálculos intermedios.

En resumen, el mapeo de proyección para la cámara PTZ puede escribirse como:

$$(u, v)^T = F_{PTZ}({}^M p; \gamma_{i,PTZ}, \gamma_{e,PTZ}) \quad (3)$$

La variable *zoom* de la cámara está relacionada con la distancia focal. Por lo tanto el proceso de calibración intrínseca debe realizarse para cada valor de *zoom* en el rango entre 0 y 100 %.

Para la calibración intrínseca se utilizan imágenes de un

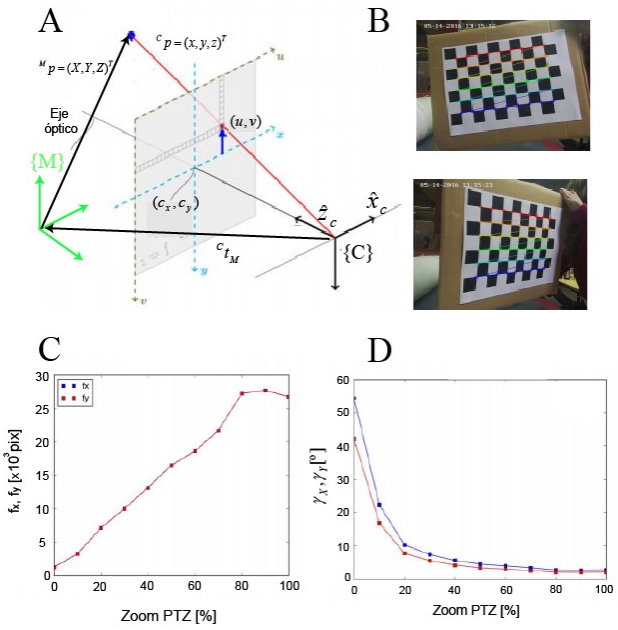

Fig. 2. (A) Representación del modelo *pinhole* para la cámara PTZ. La trama {C} representa la pose de la cámara (posición y orientación). (B) Imágenes con distintas vistas del tablero y los puntos de interés detectados durante el proceso de calibración. (C) Relación entre la distancia focal y el *zoom* en la cámara PTZ. (D) Relación entre el campo de visión (γ_X, γ_Y) en función del nivel de *zoom* en la cámara PTZ. Para valores de *zoom* altos, se notó que el equipo realizaba un movimiento de lentes para hacer foco en la imagen. Se interpretó que es este el motivo de la irregularidad en la linealidad que se observa en la Fig. 2C cuando el *zoom* es mayor a 70 %.

tablero [13], (Fig. 2B). Los pasos del procedimiento son: 1) Capturar 10 imágenes o más del tablero en distintas poses para cada valor de *zoom*. 2) Encontrar automáticamente las esquinas interiores del tablero en cada imagen mediante la función *findChessboardCorners*. Estos son los puntos que se usan para establecer la relación entre sus coordenadas 3D y las 2D en el plano de la imagen. 3) Asignar a cada punto una posición en el mundo, con Z=0 por encontrarse todos en el mismo plano. 4) Con el dataset del punto anterior (asociado una distancia focal fija), utilizar la función *calibrateCamera* para estimar la distancia focal y las poses de los tableros respecto a la cámara (que no son utilizadas en esta fase). La estimación se realiza minimizando el error de proyección sobre la imagen respecto a los parámetros antes mencionados. Se decidió que el punto principal no fuera una variable de ajuste sino que tomara los valores fijos (c_x, c_y) = (640, 480), reduciendo el número de parámetros a estimar y la incerteza en los parámetros (f_x, f_y). Los valores (f_x, f_y) estimados para cada valor de *zoom* se muestran en la Fig. 2C. Su relación con el campo de visión (γ_X, γ_Y) se obtiene a partir de:

$$\gamma_{x,y} = arctan(S_{x,y}/(2f_{x,y})) \quad (4)$$

Siendo S_x, S_y el tamaño del sensor medido en píxeles. Los resultados del campo de visión (γ_X, γ_Y) en función de la variable *zoom*, se muestran en la Fig. 2D.

B. Calibración Intrínseca de la Cámara Fisheye

El modelo de cámara *fisheye* (5-7) que se eligió para la calibración es el utilizado por la librería de OpenCV [17]. Consiste en proyectar un punto del mundo ${}^M p=(X,Y,Z)^T$ a la imagen usando el modelo *pinhole* (5) y luego, aplicar una distorsión angular para obtener la proyección final en la imagen *(u, v)* (6-7). Los parámetros intrínsecos de este modelo (que denominaremos $\gamma_{i,FE}$), son las distancias focales (f_x, f_y), el punto principal (c_x, c_y) y los coeficientes de distorsión radial (k_1, k_2, k_3, k_4). Los parámetros extrínsecos ($\gamma_{e,FE}$) son equivalentes a los descriptos en la sección anterior pero ahora con la trama {VCA} asociada a la cámara FE:

$$\begin{aligned} {}^{VCA}p &= {}^{VCA}_{M}R \cdot {}^M p + {}^{VCA}t_M \\ x' &= x/z \\ y' &= y/z \end{aligned} \quad (5)$$

Se aplica una distorsión radial de tipo polinomial en θ, obteniéndose θ_d:

$$\begin{aligned} r^2 &= x'^2 + y'^2 \\ \theta &= arctan(r) \\ \theta_d &= \theta \cdot (1 + k_1 \cdot \theta^2 + k_2 \cdot \theta^4 + k_3 \cdot \theta^6 + k_4 \cdot \theta^8) \\ x'' &= (\theta_d/r) \cdot x' \\ y'' &= (\theta_d/r) \cdot y' \end{aligned} \quad (6)$$

Donde r es el radio de la proyección (x', y') en el plano unitario. Finalmente, las coordenadas en píxeles *(u, v)* sobre el sensor CCD están dadas por:

$$\begin{aligned} u &= f_x \cdot x'' + c_x \\ v &= f_y \cdot y'' + c_y \end{aligned} \quad (7)$$

Notar que las variables *(x', y', x'', y'', r, θ, θ_d)* son variables internas del modelo solo utilizadas en cálculos intermedios.

De este modo, el mapeo de proyección para la cámara FE puede escribirse como:

$$(u, v)^T = F_{FE}({}^M p; \gamma_{i,FE}, \gamma_{e,FE}) \quad (8)$$

El método que utiliza OpenCV para la calibración intrínseca de cámaras VCA es similar al de la calibración intrínseca de la cámara PTZ, pero usando funciones de la librería *fisheye*.

Se utilizaron 10 imágenes de tableros (Fig. 3) y las estimaciones para los parámetros intrínsecos fueron: f_x=475.6, f_y=478.0, c_x=959, c_y=959, k_1=0.088, k_2=-0.023, k_3=0.024, k_4=-0,006. El campo visual de la cámara FE es fijo con un valor de 360º x 183º.

Fig. 3. Imágenes de tres vistas del tablero tomadas con la cámara *fisheye* que fueron utilizadas para la calibración intrínseca de la cámara FE.

C. Retro-Proyección al Plano de Observación

Como se explicó en la introducción, nuestro método asume que los objetos de interés se mueven sobre el plano de observación Z=0 en la trama {M}. De este modo, para apuntar la cámara PTZ solo es necesario hallar las coordenadas del punto sobre el plano ${}^{M}p=(X,Y,0)^{T}$ a partir de las coordenadas del punto en la cámara FE (u, v). Este mapeo de retro-proyección se denominó *inverse*, y se escribe como:

$$ {}^{M}p = inverse(u, v; \gamma_{i,FE}, \gamma_{e,FE}) \tag{9}$$

El procedimiento para hallar el mapeo de retro-proyección es similar al desarrollado en [4], pero ahora utilizando el modelo de distorsión (6). Esta función fue implementada en Python como parte de un conjunto de funciones de calibración de cámaras de varios modelos y que se encuentran públicas en el repositorio online GitHub [16].

IV. Calibración Extrínseca

A. Método de Calibración Extrínseca

La calibración extrínseca de las cámaras PTZ o FE, consiste en hallar los parámetros extrínsecos (γ_e) dado un dataset de calibración. Este dataset está formado por las coordenadas 3D de los puntos de calibración ${}^{M}p_i'$ en el mundo y las mediciones de sus proyecciones en la cámara (u_i', v_i'), (el índice i representa el número de medición en el dataset). La función *solvePnP* de OpenCV estima los parámetros extrínsecos γ_e minimizando el error de proyección sobre la imagen ε_I descripto por:

$$ \varepsilon_I = \sum_i \left| F({}^{M}p_i'; \gamma_i, \gamma_e) - (u_i', v_i')^{\mathrm{T}} \right|^2 \tag{10}$$

Donde F representa el mapeo de proyección de las cámaras PTZ o FE (3, 8). En OpenCV, $\gamma_e = \{\mathbf{r}, \mathbf{t}\}$, donde el vector **r** representa el eje y el ángulo de rotación (asociado a ${}^{C}_{M}R$) y el vector **t,** corresponde a ${}^{C}t_M$.

El primer testeo de calibración extrínseca para ambas cámaras se realizó en condiciones de laboratorio utilizando una foto de un tablero. Se asignaron coordenadas 3D que respetaban el tamaño real de los cuadrados: 0.0374 m. Los pasos que se desarrollaron fueron: 1) Tomar una imagen del tablero. 2) Detectar automáticamente las esquinas internas con la función *findChessboardCorners* y usarlas como puntos de calibración. 3) Asignar a cada esquina su posición en el marco de coordenadas del mundo (tablero en Z=0). 4) Ejecutar la función *solvePnP* usando el conjunto de píxeles del punto 2), el conjunto de posiciones del punto 3), y los parámetros intrínsecos estimados previamente. Una vez estimado el parámetro **r**, se construyó la matriz ${}^{C}_{M}R$ con la función *Rodrigues* [13], [14] de OpenCV.

B. Resultados de la Calibración Extrínseca

Cámara PTZ: Se colocó la cámara PTZ a aproximadamente 1 metro de distancia del tablero, apuntándolo con *zoom* = 0 (Fig. 4A). La pose estimada fue: **r** = (-0.04, 2.93, -0.41) y **t** = (0.10, -0.29, 1.26). El módulo de **t** obtenido fue 1.29 m, que es un resultado coherente con el entorno de prueba descripto. Para la verificación de los resultados se proyectaron las posiciones 3D de los puntos de calibración y se compararon con las esquinas detectadas automáticamente. Para la proyección se utilizó la funcion *projectPoints* de la librería OpenCV que aplica el mapeo (3). La comprobación se muestra gráficamente en la Fig. 4B donde se confirma que los parámetros extrínsecos adquiridos son correctos. En azul se grafican los puntos de las esquinas detectadas por el algoritmo de calibración extrínseca y en rojo los puntos resultantes de proyectar la grilla de 0,0347m sobre la imagen.

Cámara FE*:* Los pasos del algoritmo para estimar los

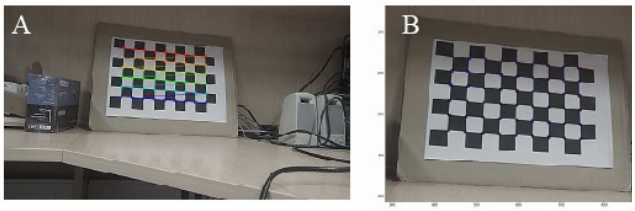

Fig. 4. (A) Imagen del tablero y los puntos de interés detectados con los que se ejecuta el algoritmo de calibración extrínseca. (B) Resultado de la función *projectPoints*.

parámetros extrínsecos son los mismos que se describieron en la sección anterior para la cámara PTZ. La función *solvePnP* se aplicó utilizando los parámetros intrínsecos estimados para la cámara FE ($\gamma_{i,FE}$) y se optimizó el error de proyección ε_I utilizando el mapeo de la cámara FE (8).

Para testear la estimación, se utilizó la foto de un tablero (a una distancia aproximada de 1 metro) como muestra la Fig. 5A. Los resultados obtenidos para la pose fueron: **r** = (-1.26, -2.03, 0.39) y **t** = (0.10, -0.15, 0.79).

Para comprobar la estimación, se proyectaron los puntos 3D (${}^{M}p_i$) calculados con (8) para compararlos con las esquinas detectadas por OpenCV. En la Fig. 5B los puntos azules representan las esquinas detectadas por el algoritmo de calibración extrínseca y los rojos, los puntos resultantes de proyectar los puntos 3D sobre la imagen.

Además, usando (9) se realizó una retro-proyección de las esquinas detectadas para contrastar la predicción de las posiciones sobre el plano mundo (Z=0). Este último chequeo es el más significativo y se realiza para garantizar que los parámetros extrínsecos que se estimaron permitan transformar (u,v) en la cámara *fisheye* {FE} al plano mundo {M} con el menor error posible, ya que en nuestro método, éste es el primer paso necesario para un apunte correcto de la cámara PTZ.

Los resultados obtenidos para la retro-proyección sobre el plano de observación (Z=0) se muestran en las Fig. 5C. En azul se muestran los puntos 3D asignados a las esquinas detectadas por el algoritmo de calibración extrínseca y en rojo los puntos retroproyectados de la función *inverse* (9).

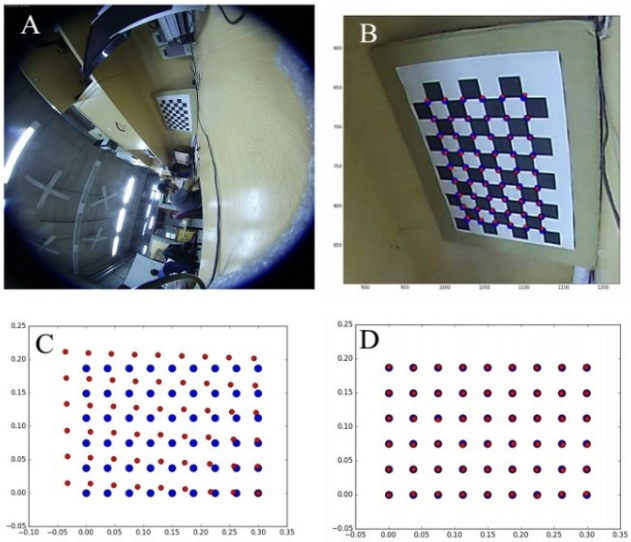

Fig. 5. (A) Fotograma del tablero donde se detectan los puntos de interés para la calibración extrínseca con *solvePnP*. (B) Acercamiento mostrando en azul las esquinas internas detectadas y en rojo el resultado de la función *projectPoints* con la calibración extrínseca obtenida por *solvePnP*. (C) En azul las coordenadas mundo reales asignados a las esquinas del tablero y en rojo los puntos retro-proyectados de la función *inverse* con los mismos parámetros extrínsecos que en (B). (D) Comprobación con *inverse* usando los vectores **r** y **t** optimizados sobre el plano de observación.

El error que se observa en la proyección de puntos en la imagen (Fig. 5B) es despreciable frente al que se observa en la retro-proyección al mundo (Fig. 5C). Esto se debe a que se utilizó el método tradicional que utiliza la función *solvePnP* ejecutándose una minimización del error ε_I sobre la imagen, expresado en (10). Se utilizaron varias imágenes para testear la calibración y con ellas aparecieron errores significativos. A partir del error que se percibe gráficamente se decide realizar una optimización de los valores de **r** y **t** que minimicen el error de retro-proyección. En la próxima sección se presenta un método para corregir este problema.

C. Optimización de la Retro-Proyección Disminuyendo el Error de Predicción en el Plano de Observación

A diferencia del enfoque tradicional que minimiza el error de proyección en el espacio imagen (10), en este trabajo proponemos minimizar el error cuadrático que existe entre las coordenadas 3D resultantes de la función *inverse* y los puntos 3D medidos ${}^{M}p_i'$, que definimos como ε_M:

$$\varepsilon_M = \sum_i \left| inverse(u_i', v_i'; \gamma_{i,FE}, \gamma_{e,FE}) - {}^{M}p_i' \right|^2 \qquad (11)$$

Para la optimización se utilizaron como condición inicial los **r** y **t** obtenidos de la calibración extrínseca tradicional con *solvePnP*. Luego, se minimizó ε_M utilizando la función *minimize* del paquete estándar *scipy.optimize* de Python [20]. El error de retro-proyección ε_M optimizando los parámetros extrínsecos $\gamma_{e,FE}$ según el enfoque tradicional (que minimiza ε_I) fue $7.2 \cdot 10^{-4}$ m^2. Agregando la minimización adicional el error de retro-proyeccion ε_M (11) disminuyó a $8.2 \cdot 10^{-7}$ m^2. En la Fig. 5D, puede observarse que luego de la optimización de ε_M (11) los resultados de la retro-proyección mejoran sustancialmente.

V. Apunte de la Cámara PTZ en Función de un Punto de Interés Observado en la Cámara FE

A. Geometría Asociada al Apunte de la Cámara PTZ

El método de apunte propuesto en este trabajo se inicia con una captura desde la cámara FE luego de ubicar un objeto de interés sobre el plano de observación. A continuación, se deben hallar los ángulos para apuntar la cámara PTZ de forma tal de que el objeto quede centrado en la imagen (ver Fig. 6). Finalmente se calcula el campo de visión y se modifica el nivel de *zoom* para que el objeto se vea en mayor detalle.

En esta sección, utilizaremos una notación compacta para

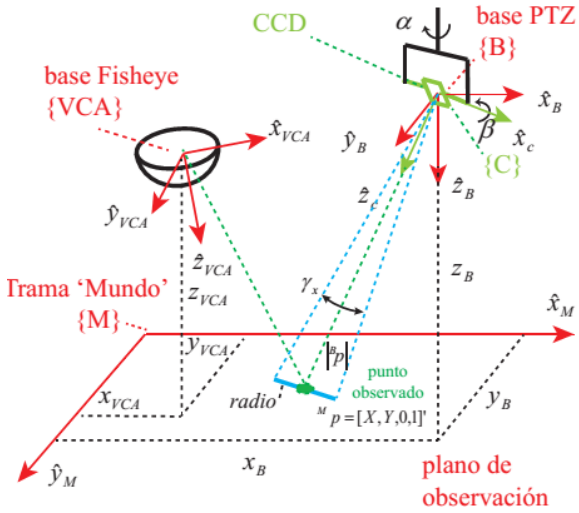

Fig. 6. Geometría asociada al problema de apunte de la cámara PTZ a partir de la detección de un punto de interés en la imagen de la cámara *fisheye*. Las tramas asociadas al problema son: {M}, trama Mundo; {VCA}, base fija de la cámara FE; {B} base fija de la cámara PTZ; {C} trama giratoria (*pan-tilt*) asociada al eje óptico de la cámara PTZ. Los ángulos (α, β) corresponden a los movimientos de *pan* y *tilt* respectivamente. Las líneas punteadas de color celeste representan el campo de visión de la cámara PTZ.

las transformaciones de coordenadas en las ecuaciones (1,5). Para esto utilizaremos coordenadas homogéneas [14], de forma tal que (1) puede escribirse como: ${}^{C}p = {}^{C}_{M}T \cdot {}^{M}p$, con ${}^{C}p = (x, y, z, 1)^T$ y ${}^{M}p = (X, Y, Z, 1)^T$. La matriz homogénea T, está dada por:

$${}^{C}_{M}T = \begin{bmatrix} {}^{C}_{M}R & {}^{C}t_M \\ 0_{1\times3} & 1 \end{bmatrix} \qquad (12)$$

Conocidos los parámetros intrínsecos y extrínsecos de la cámara *fisheye* VCA y el pixel *(u,v)* observado, se aplica la retro-proyección (9), calculándose ${}^{M}p = (X, Y, 0, 1)^T$ [4]. Conocida la calibración de la cámara PTZ, se calculan las coordenadas del punto de observación en la trama {B} asociada a la base fija de la cámara PTZ, ${}^{B}p = (X_B, Y_B, Z_B, 1)^T$ (Fig. 6). Se calculan los ángulos de *pan* y *tilt* (α, β respectivamente) apuntando hacia un objeto y finalmente se

elige el nivel de *zoom* para observarlo completamente (ver Fig. 6).

La pose de la cámara PTZ ${}^{C}_{M}T$, se obtuvo en la calibración extrínseca que se realizó en la sección anterior. De este modo, es posible calcular la pose de la trama {M} respecto a la base de la cámara PTZ {B}, como:

$$ {}^{B}_{M}T = {}^{B}_{C}R(\alpha, \beta) \cdot {}^{C}_{M}T \tag{13} $$

La matriz de rotación representa la rotación del mecanismo con un ángulo α asociado al *pan* y β al *tilt* y está dada en coordenadas homogéneas por (14):

$$ {}^{B}_{C}R(\alpha, \beta) = \begin{bmatrix} c(\alpha) & -s(\alpha)c(\beta) & -s(\alpha)s(\beta) & 0 \\ s(\alpha) & c(\alpha)c(\beta) & c(\alpha)s(\beta) & 0 \\ 0 & -s(\beta) & c(\beta) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{14} $$

Por lo tanto, se puede calcular ${}^{B}_{M}T$ del siguiente modo durante el proceso de calibración: 1) Se captura una imagen con la cámara PTZ en la que se eligen puntos de calibración; 2) Se obtienen los ángulos *pan* y *tilt* en los que se encuentra posicionada la cámara PTZ al momento de la captura con la función *getStatus* del protocolo ONVIF. 3) Se arma la matriz de rotación (14). 4) Se ejecuta el método de calibración extrínseca usando la imagen del punto 1). 5) Se calcula ${}^{B}_{M}T$ con (13). Luego, se calculan las coordenadas del punto de interés respecto a {B} ${}^{B}p = {}^{B}_{M}T \cdot {}^{M}p$ a partir de las coordenadas respecto al mundo ${}^{M}p$ obtenidas con la retro-proyección desde cámara VCA al plano mundo (9).

La ecuación que determina el apunte de la cámara PTZ al punto de observación está determinada por:

$$ {}^{C}p = (0,0,|{}^{B}p|,1)^{T} = {}^{C}_{B}R(\alpha, \beta) \cdot {}^{B}p \tag{15} $$

Donde $|{}^{B}p| = \sqrt{{X_B}^2 + {Y_B}^2 + {Z_B}^2}$. Esto da lugar a un conjunto de ecuaciones que permiten hallar los ángulos α y β:

$$ \alpha = arc\tan2(-X_B, Y_B) \tag{16} $$
$$ \beta = arc\tan2(({X_B}^2 + {Y_B}^2), Z_B) $$

B. Prueba del Sistema en un Ambiente Interior

Se colocaron las cámaras en un soporte construido para hacer las pruebas de integración a una altura de 3 m (Fig. 7A). En el suelo se marcaron y enumeraron los puntos que luego fueron usados para la calibración extrínseca de ambas cámaras. Se fijó el origen de coordenadas en el mundo y se les asignaron posiciones *(X, Y, 0)*. Los métodos se ejecutaron con las coordenadas en píxeles donde se encontraban los puntos en las dos imágenes y sus posiciones reales en el mundo. Las capturas de las cámaras se muestran en las Fig. 7B-C. En la Fig. 7C, se comprueban los parámetros de la PTZ usando *projectPoints* y chequeando las coordenadas en píxeles de los puntos de interés con la proyección de las posiciones reales del mundo en la imagen. En la Fig. 7D se muestra la verificación de los valores de los parámetros extrínsecos de la cámara FE a través del método *inverse* donde se comparan las coordenadas de las posiciones reales y la retro-proyección de los puntos de la imagen al mundo.

Para testear la geometría de apunte, se colocó un objeto en una posición que no pertenezca al dataset de calibración y se ejecutó el método con esas coordenadas. La imagen de la cámara PTZ luego de rotar los ángulos *pan* y *tilt* resultantes se muestra en la Fig. 7E-F.

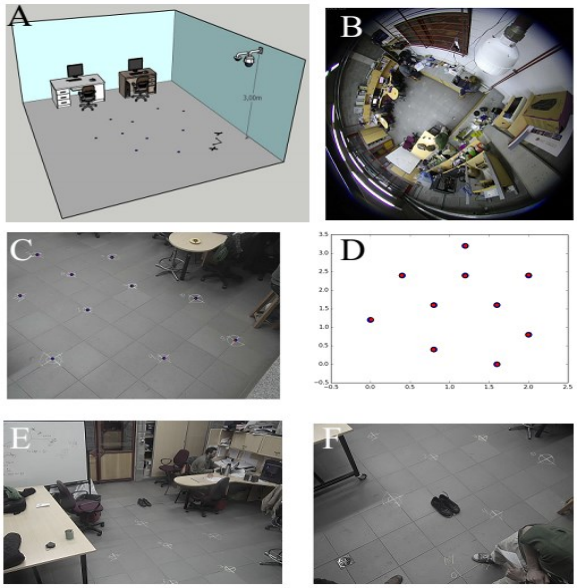

Fig. 7. (A) Entorno de prueba. (B) Captura de cámara *fisheye* para calibración extrínseca. (C) Captura de cámara PTZ para calibración extrínseca. (D) Azul: representa los datos reales. Rojo: representa el resultado de las funciones de verificación. (E-F) Captura de cámara PTZ luego de rotar los ángulos *pan* y *tilt.*

Para estimar el error de apunte, se apuntó la cámara PTZ hacia puntos del plano que no fueron utilizados durante el proceso de calibración. Luego del apunte automático de la PTZ, se movieron los motores utilizando la interfaz de usuario hasta que el punto de interés quedó ubicado en el centro de la captura de la cámara PTZ. Con la función ***getStatus*** de ONVIF se obtuvo el valor de *pan* y *tilt* en la que se encontraba eje óptico y se comparó con los que habían resultado del apunte automático. Durante el período de testeo de este método se utilizaron varios entornos de prueba en los que se modificaban las posiciones de las cámaras y de los puntos de interés. El error máximo obtenido fue de 3.9° en el ángulo de *pan* y 5.5° en el ángulo de *tilt.* El error promedio fue de 5º (n=20 mediciones).

C. Ajuste de Zoom

La última fase del desarrollo consistió en definir una estrategia para determinar el nivel de *zoom*. La relación que existe entre el *zoom* y el campo de visión se obtuvo durante la calibración intrínseca de la cámara (Fig. 2D). Para calcular el ángulo del campo de visión que se necesita, se plantea la situación mostrada en la Fig. 6, donde el objeto de interés se encuentra a una distancia $|{}^{B}p|$ de la cámara. Se decide qué radio alrededor del objetivo se quiere incluir en la imagen y se calcula el ángulo de visión γ_x usando (17). La relación propuesta es:

$$ \tan(\gamma_x/2) = radio/|{}^{B}p| \tag{17} $$

Donde *radio* se considera conocido ya que puede ser elegido por el operador (a través de un conocimiento del rango de tamaños de los objetos de interés). En el sistema, se calcula $\left|{}^{B}p\right| = \sqrt{{X_B}^2 + {Y_B}^2 + {Z_B}^2}$ a partir de la coordenada en la que se encuentra el objeto en el mundo (ver sección anterior).

D. Prueba del Sistema en un Ambiente Exterior

Se instaló el sistema FE+PTZ en un ambiente exterior en condiciones típicas de funcionamiento (Fig. 8A). Se realizó la calibración del sistema. Se tomó una captura con cada una de las cámaras. En las Fig. 8B-C se muestran las imágenes utilizadas para la calibración y los puntos claramente definidos en cada una de ellas. Se indica en rojo el origen de las coordenadas del marco de referencia del mundo {M}. La disposición de las losas en el piso se tomó como una grilla de calibración (midiéndose las distancias entre el los para asignar sus posiciones en el mundo). Para ejecutar las calibraciones de las dos cámaras se usó un conjunto de 6 puntos.

E. Comprobación de la Calibración Extrínseca

Con el procedimiento de calibración también se puede obtener la posición de las cámaras en el mundo. Estos datos fueron utilizados como un método adicional de verificación

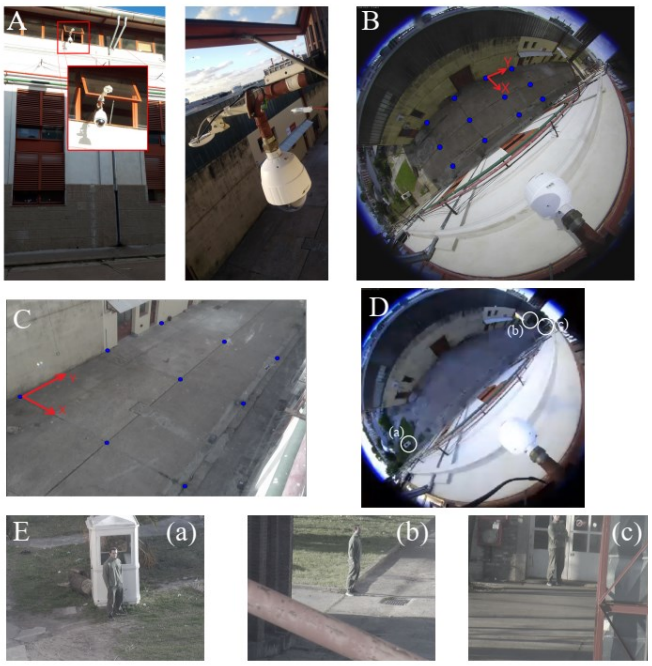

Fig. 8. (A) Soporte con cámaras instaladas. (B) Captura de la cámara *fisheye* con los puntos de calibración. (C) Captura de la cámara PTZ con los puntos de calibración. (D) Imagen de la cámara *fisheye*. Las regiones (a-d) representan distintas regiones de interés indicadas por el operador. (E) Imágenes adquiridas a través del posicionamiento y *zoom* automático de la cámara PTZ.

del algoritmo. Se midió la posición del soporte con los equipos con una cinta métrica y se obtuvo ${}^{M}t$ = (9.88, -6.0, 8.65) metros. Los vectores **r** y **t** estimados para la cámara PTZ fueron: **r** = (-0.82, 2.60, 1.23) y **t** = (5.96, 0.04, 12.760; y para la cámara VCA: **r** = (2.81, 1.38, 0.10) y **t** = (-1.60, -10.77, 8.49). Para calcular las posiciones de las cámaras VCA y PTZ (que se encuentran en los orígenes de las tramas {VCA} y {C} respectivamente) se invierten las matrices ${}^{VCA}_{M}T$ y ${}^{C}_{M}T$ obteniendo: ${}^{M}t_{VCA}$ = (9.04, -5.5, 8.83), ${}^{M}t_{C}$ = (9.52, -5.4, 8.87).

La diferencia que existe entre estas posiciones en los ejes x e y, se debe a que el soporte desde la pared a la PTZ tiene 0.45 m y de la PTZ a la FE 0.4m. Se puede ver que la diferencia entre las medidas tomadas y los resultados del cálculo es muy pequeña, lo que ratifica una vez más el método de calibración.

Finalmente se testeó el sistema de apunte de la cámara PTZ para distintas direcciones de observación indicadas en la imagen *fisheye* (ver Fig. 8D). En la Fig. 8E se puede observar el funcionamiento del sistema de apunte y ajuste de *zoom* (con *radio*= 2m). Los resultados son satisfactorios, pues en todos los casos, se logra ver el objetivo de apunte en la captura final.

VI. Discusión y Conclusiones

En este trabajo se propone un sistema compuesto por una cámara *fisheye* (que monitorea un hemisferio completo de la escena) y una cámara PTZ, que captura imágenes de alta resolución de regiones de interés. Una suposición razonable en los sistemas de monitoreo urbano, es que los objetos de interés (vehículos y peatones), se mueven sobre el plano terrestre [6]. Esta suposición simplifica notablemente el problema estimación de la posición 3D de los objetos de interés, ya que no es necesario utilizar algoritmos de coincidencia estéreo como los aplicados en [21]-[22]. Sin embargo, el trabajo [6] tiene la limitante de asumir que las bases de las cámaras VCA y PTZ son perpendiculares a la superficie de observación.

El aporte principal de nuestro trabajo es relajar esta suposición, desarrollando una metodología de calibración y de retro-proyección que es válida para el caso general de poses arbitrarias en ambas cámaras.

La calibración intrínseca de las cámaras se realizó con éxito utilizando la librería de software libre OpenCV [17]-[19] y es importante destacar que el modelo (5-7) puede ser utilizado en otras cámaras VCA (dióptricas o catadióptricas).

Para resolver el problema de retro-proyección, se desarrolló la función *inverse* (9) para cámaras FE [16], mostrándose que se obtiene una mejora sustancial en la predicción de posición sobre el plano de observación optimizando el error de retro-proyección ε_M (11), (Fig. 5).

Para garantizar la flexibilidad e interoperabilidad del sistema (sin importar la marca de los dispositivos) se utilizó el protocolo libre ONVIF que permite acceder, configurar y controlar las cámaras FE y PTZ de distintos fabricantes.

Se desarrolló una estrategia de apunte de la cámara PTZ a partir de un punto elegido en la imagen de la cámara FE, obteniéndose un error promedio de 5º (con 6 puntos de calibración). Este resultado es similar al obtenido con el método [6] (con 96 puntos de calibración). Consideramos que esta diferencia puede deberse a que el método [6] minimiza el error en el espacio imagen y esto aumenta el error en la localización del objeto sobre el plano de observación (Fig. 5).

Las soluciones comerciales actuales de sistemas FE+PTZ [23] imponen restricciones importantes tanto en la altura, orientación y distancias entre cámaras. Por otro lado, necesitan de más de 50 puntos de calibración para poder ajustar el

método de posicionamiento de la cámara PTZ en función de la imagen FE. La solución planteada en este trabajo tiene mayor flexibilidad en relación a la pose de las cámaras, y como se muestra en la Fig. 8, presenta resultados satisfactorios con 6 puntos de calibración.

Se propuso un método simple para la estimación del grado de *zoom* basado en el tamaño característico del objeto que se desea observar (17). Consideramos que la dependencia de un valor *a priori* para el parámetro (*radio*), no es una limitante grave del método. Esto se debe a que en escenas urbanas típicas, el rango de tamaños de los objetos de interés (peatones y vehículos) está acotado. A futuro, otra forma de mejorar esta estrategia podría ser clasificar el objeto y luego determinar el nivel de *zoom* en función su tamaño característico o calcular su tamaño angular en función de la dimensión angular del objeto en la imagen FE y a partir de ahí, determinar óptimamente el nivel de *zoom* [2].

En trabajos previos [3]-[4] abordamos la problemática de detección de vehículos en movimiento, geo-localización y estimación de velocidad a partir de imágenes obtenidas con cámaras FE. Sin embargo, para lograr un algoritmo automático robusto que controle satisfactoriamente el apunte de la cámara PTZ frente a cambios drásticos en las condiciones climatológicas y de luminosidad, es necesario avanzar en el entrenamiento de clasificadores de objetos que sean precisos para este tipo de imágenes con fuertes distorsiones.

REFERENCIAS

[1] G. H. Minari. Anomalies Identification in Images from Security Video Cameras Using Mask R-CNN. *IEEE Latin America Transactions*, *18*(03), pp. 530-536, 2020.

[2] D. Stanganelli, D. E. Oliva, M. Noblía, and F. Safar. Calibración de una cámara fisheye comercial con el modelo unificado para la observación de objetos múltiples. In *2014 IEEE Biennial Congress of Argentina (ARGENCON)*, pp. 147-152, 2014.

[3] S. I. Arroyo, F. Safar, and D. Oliva. Probabilidad de infracción de velocidad de vehículos utilizando visión artificial en cámaras de campo amplio. In *2016 IEEE Biennial Congress of Argentina (ARGENCON)*, pp. 1-6, 2016.

[4] S. I. Arroyo, U. Bussi, F. E. Safar and D. Oliva. A monocular wide-field vision system for geolocation with uncertainties in urban scenes. *Engineering Research Express*, 2, pp 1-20,2020.

[5] V. B. de Oliveira, R. Barth, Oliveira, M. A. de Oliveira and V. E.. Nascimento. Vehicle speed monitoring using convolutional neural networks. *IEEE Latin America Transactions*, *17*(06), 1000-1008, 2019.

[6] C. H. Chen. Heterogeneous fusion of omnidirectional and PTZ cameras for multiple object tracking. *IEEE Transactions on Circuits and Systems for Video Technology*, *18*(8), 1052-1063, 2018.

[7] Y. Bastanlar. A simplified two-view geometry based external calibration method for omnidirectional and PTZ camera pairs. *Pattern Recognition Letters*, *71*, 1-7, 2016.

[8] C. Cai, B. Fan, X. Liang and Q. Zhu,. Automatic Rectification of the Hybrid Stereo Vision System. *Sensors*, *18*(10), 3355, 2018.

[9] ONVIF Contributors. Página oficial onvif. https://www.onvif.org/, June 2020.

[10] ONVIF Contributors. Especificaciones del core. http://www.onvif.org, June 2020.

[11] ONVIF Contributors. ONVIF Application Programmer's Guide. Addison-Wesley, 2011.

[12] Ltd. Quatanium Co. python-onvif-zeep. https://github.com/FalkTannhaeuser/python-onvif-zeep, September 2018. Github repository licensed under the MIT License.

[13] G. Bradski and A. Kaehler. Learning OpenCV: Computer vision with the OpenCV library. O'Reilly Media, Inc., 2019.

[14] J. J. Craig. Introduction to robotics: mechanics and control, 3/E. Pearson Education India, 2009.

[15] P. Corke. Robotics, vision and control: fundamental algorithms in MATLAB® second, completely revised (Vol. 118). Springer, 2017.

[16] S. I. Arroyo. Repositorio github. https://github.com/sebalander /sebaPhD /tree/master/calibration, June 2020.

[17] OpenCV documentation. Fisheye camera model, https://docs.opencv.org /4.4.0, June 2020.

[18] OpenCV documentation. Rodrigues. https://docs.opencv.org/4.4.0, June 2020.

[19] OpenCV documentation. Camera Calibration and 3D Reconstruction. https://docs.opencv.org/4.4.0, June 2020.

[20] P. Virtane, SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nature methods*, *17*(3), 261-272, 2020.

[21] H. Y. Lin, , M. L. Wang. HOPIS: Hybrid omnidirectional and perspective imaging system for mobile robots. *Sensors*, *14*(9), pp. 16508-16531, 2014.

[22] M. Rostkowska, P. Skrzypczyński. Hybrid field of view vision: From biological inspirations to integrated sensor design. In *2016 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, pp. 629-634, 2016.

[23] VIVOTEC Panoramic PTZ Installation Guide. https://www.vivotek.com/panoramic%20ptz#downloads.

[24] Z. Zhang. A flexible new technique for camera calibration. *IEEE Transactions on pattern analysis and machine intelligence*, *22*(11), 1330-1334, 2000.

Sebastian I. Arroyo. Recibió el título de Licenciado en Ciencias Físicas en el Instituto Balseiro, Universidad Nacional de Cuyo en 2013. Desde entonces es estudiante de doctorado en Ciencia y Tecnología de la Universidad Nacional de Quilmes con beca CONICET. Desde 2019 es Docente-Investigador con dedicación exclusiva en dicha casa.

Lilian García. Recibió el título de Ingeniera en Automatización y Control Industrial de la Universidad Nacional de Quilmes. Actualmente se desempeña en el sector privado.

Félix E. Safar. Recibió el título de Ingeniero en Telecomunicaciones en la Universidad Nacional de La Plata y Master of Science in Electrical Engineering en Virginia Tech. Actualmente es docente investigador en la Universidad Nacional de Quilmes y la Universidad Nacional de La Plata. Sus áreas de interés son Visión e Inteligencia Artificial, IoT y Sistemas Embebidos.

Damián E. Oliva. Recibió el título de Licenciado en Ciencias Físicas en el Instituto Balseiro, Universidad Nacional de Cuyo en 2001 y el grado de Doctor en Ciencias Biológicas (Neurociencias) de la Universidad de Buenos Aires en 2010. Actualmente es investigador del CONICET y docente en la carrera de Ingeniería en Automatización y Control Industrial de la Universidad Nacional de Quilmes. Sus áreas de interés son la Neurociencia computacional, la Visión e Inteligencia Artificial y la Robótica bioinspirada en ambientes no estructurados.