

Infraestructura de programación de robots aéreos y aplicaciones visuales con aprendizaje profundo

Pedro Arias Pérez

David Martín Gómez
José María Cañas Plaza (URJC)

Índice

1

Introducción

Problema, objetivos y material

2

Infraestructura

DroneWrapper, TelloDriver y VictureDriver

3

Aplicaciones

Sigue-color y sigue-persona

4

Conclusiones

Conclusiones y líneas futuras

1

Introducción

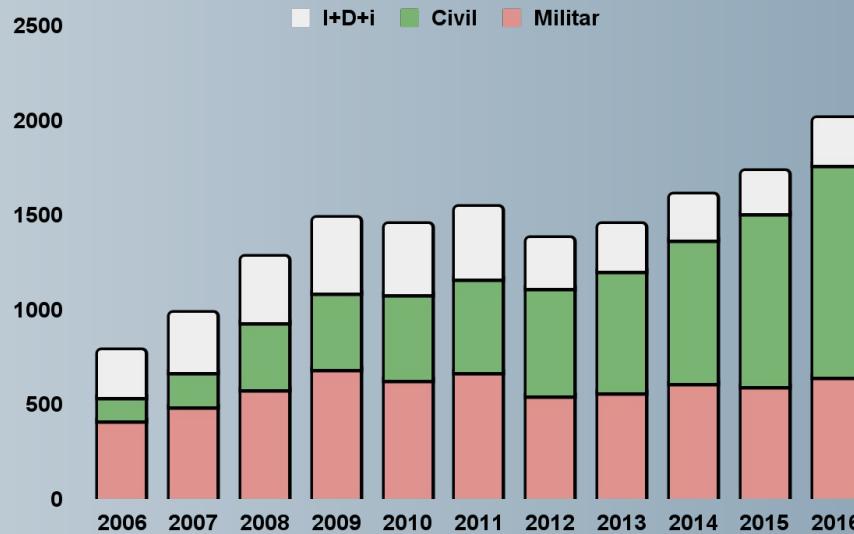
Problema, objetivos y material

Introducción



Motivación

Uso en aumento
Múltiples aplicaciones



Evolución del número de modelos de drones según su ámbito de aplicación a nivel mundial

Fuente: "Plan Estratégico para el desarrollo del sector civil de los drones en España 2018-2021". Ministerio de Fomento.

Introducción



Motivación

Uso en aumento
Múltiples aplicaciones

	2035
--	------

51.400	1.220M	11.000
Flota drones uso profesional	Volumen de negocio anual	Puestos de trabajo (ud)

	2050
--	------

53.500	1.520M	11.500
Flota drones uso profesional	Volumen de negocio anual	Puestos de trabajo (ud)

Fuente: "Plan Estratégico para el desarrollo del sector civil de los drones en España 2018-2021". Ministerio de Fomento.

Introducción



Motivación

Uso en aumento
Múltiples aplicaciones



Problema

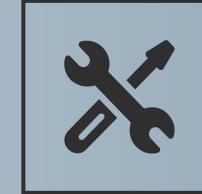
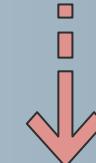
Sistemas complejos
Carencia de htas



Objetivos

Htas para la
programación y
navegación de
multicópteros

Carencia → Necesidad



Herramienta para
la programación y
navegación de
multicópteros



Multiples
plataformas aéreas



Varias aplicaciones
de visión

Material HW

Drones:



Iris Sim



DJI Tello



PX4

Jetson AGX
Xavier



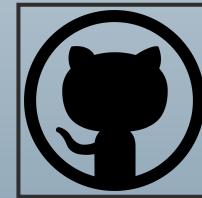
Cámara
Victure



Material SW



JdeRobot/drones



RoboticsLabURJC/2021
-tfm-pedro-arias

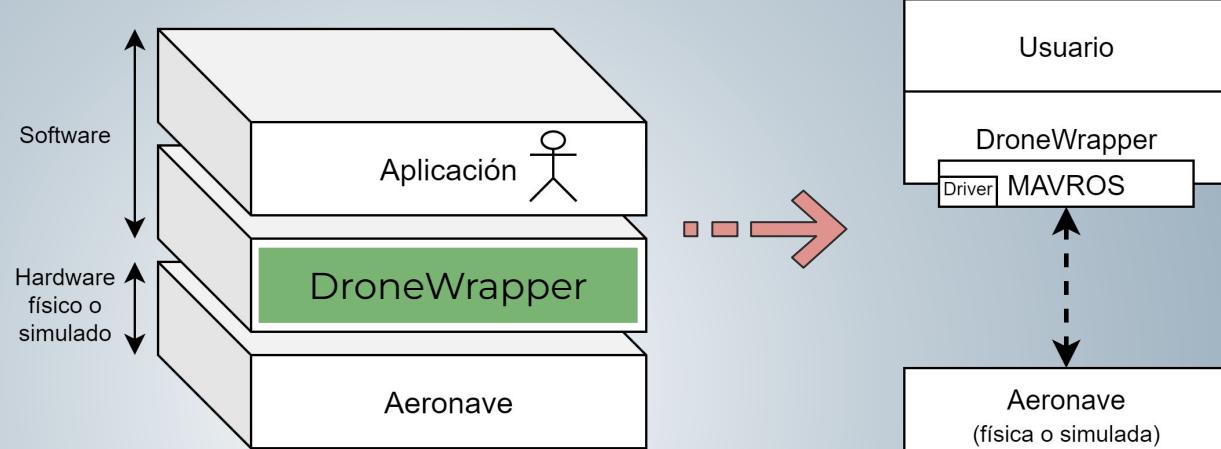
2

Infraestructura

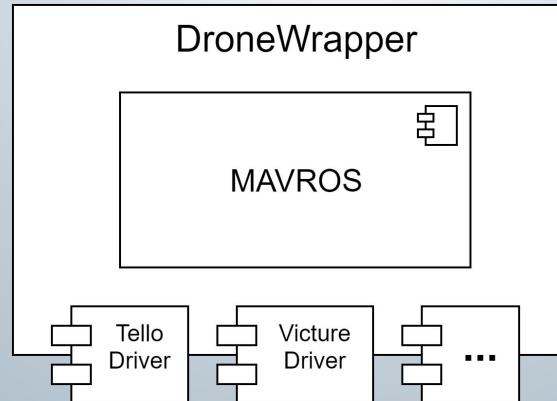
DroneWrapper
TelloDriver
VictureDriver

Diseño

Esquema de capas

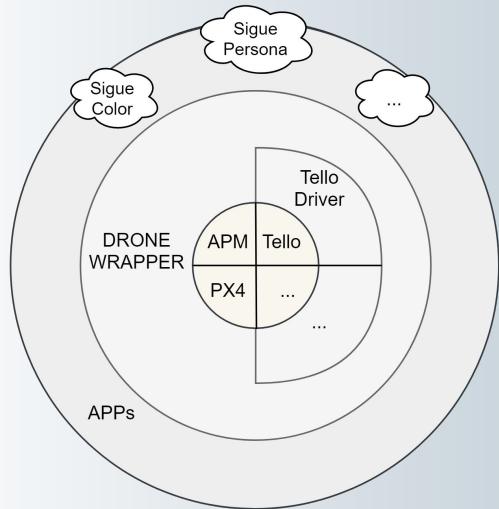


Arquitectura modular



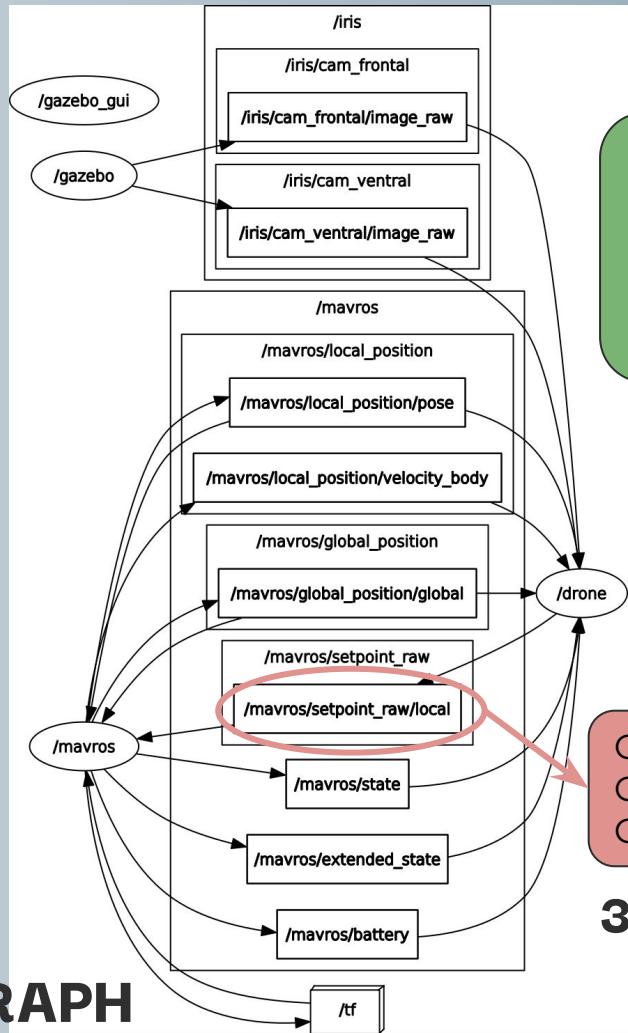
- DroneWrapper
- TelloDriver
- VictureDriver

DroneWrapper



**“Envoltorio
de drones”**

ROSGRAPH



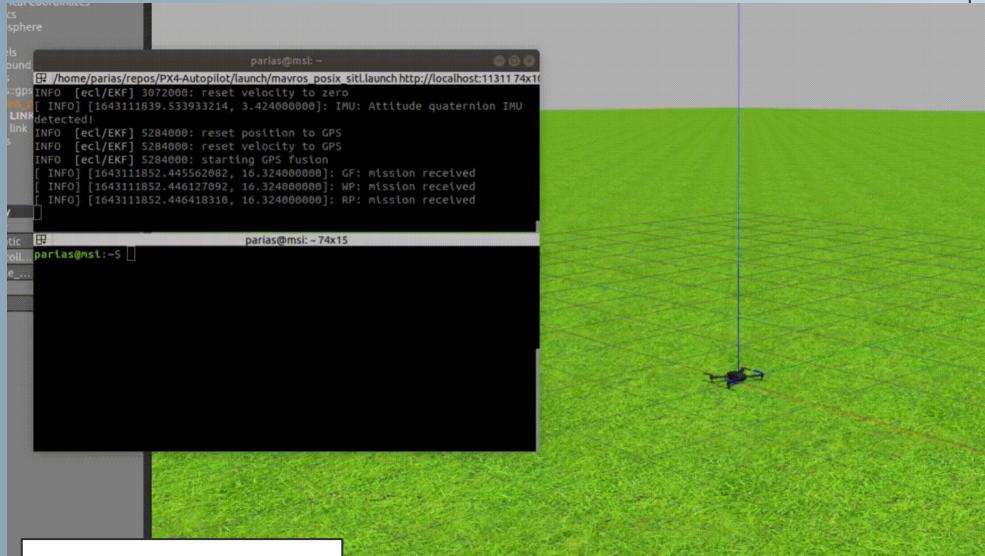
Paquete ROS

- 1 Nodo
- 8 Sub Topics
- 1 Pub Topic
- 5 Servicios
- 1 Parametro

Control por posición
Control por velocidad
Control mixto

3 tipos de control

DroneWrapper II



Estado	[x, y, z] = get_position()
	[vx, vy, vz] = get_velocity()
	rate = get_yaw_rate()
	[r, p, y] = get_orientation()
Control	s = get_landed_state()
	takeoff(h)
	land()
	set_cmd_pos(x, y, z, yaw)
Sensores	set_cmd_vel(vx, vy, vz, vyaw)
	set_cmd_mix(vx, vy, z, vyaw)
	img = get_frontal_img()
	img = get_ventral_img()

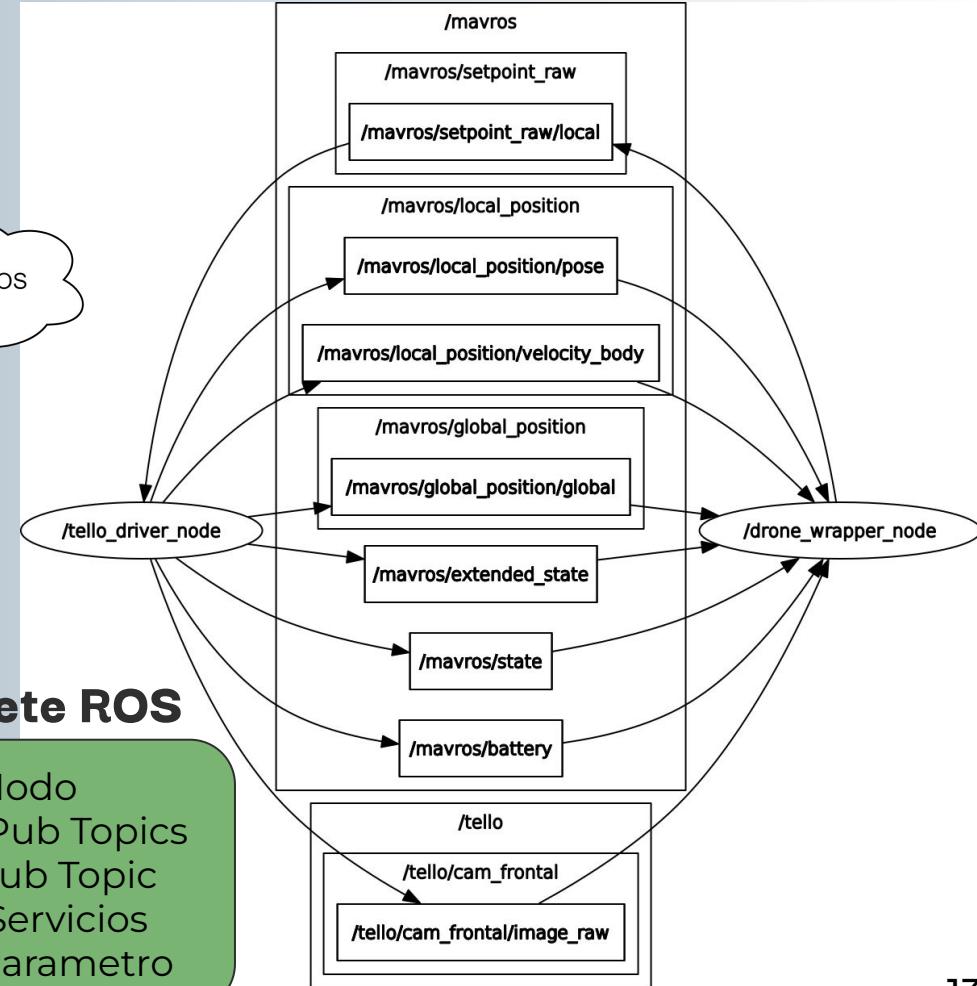
API

TelloDriver

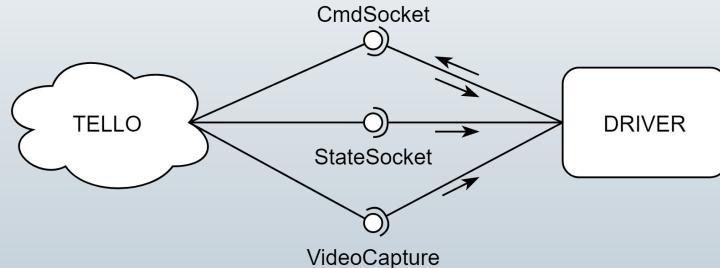


MAVROS

TELLO



TelloDriver II



Socket	Hilo	Uso
CmdSocket	Hilo ppal.	Envío comandos
	Response handler	Recepción de respuestas
StateSocket	State handler	Recepción de datos de estado
VideoCapture	Video handler	Recepción de vídeo

→ 3 Sockets

→ Hilo ppal + 3 Handlers

Tello SDK

Read Commands (xxx?)

- Returns the current value of the sub-parameters.

Tello Commands

Control Commands

Command	Description	Possible Response
Command	Enter SDK mode.	
takeoff	Auto takeoff.	
land	Auto landing.	
streamon	Enable video stream.	
streamoff	Disable video stream.	
emergency	Stop motors immediately.	
up x	Ascend to "x" cm. "x" = 20-500	
down x	Descend to "x" cm. "x" = 20-500	
left x	Fly left for "x" cm. "x" = 20-500	
right x	Fly right for "x" cm. "x" = 20-500	
forward x	Fly forward for "x" cm. "x" = 20-500	
back x	Fly backward for "x" cm. "x" = 20-500	
cw x	Rotate "x" degrees clockwise. "x" = 1-360	
ccw x	Rotate "x" degrees counterclockwise. "x" = 1-360	
flip x	Flip in "x" direction. "l" = left "r" = right "f" = forward "b" = back	ok / error
go x y z speed	Fly to "x" "y" "z" at "speed" (cm/s). "x" = -500-500 "y" = -500-500 "z" = -500-500 "speed" = 10-100	

Note: "x", "y", and "z" values can't be set between -20 – 20 simultaneously.

VictureDriver

Paquete ROS

- 1 Nodo
- 1 Pub Topic



```
cap = cv2.VideoCapture()
while:
    ret, frame = cap.read()
    if ret:
        pub.publish(frame)

cap.release()
```

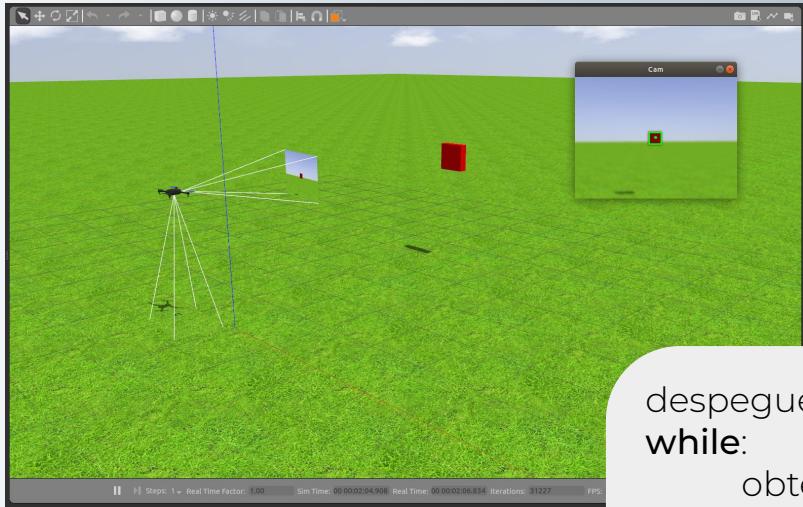
3

Aplicaciones

Sigue-color

Sigue-persona

Sigue-Color I

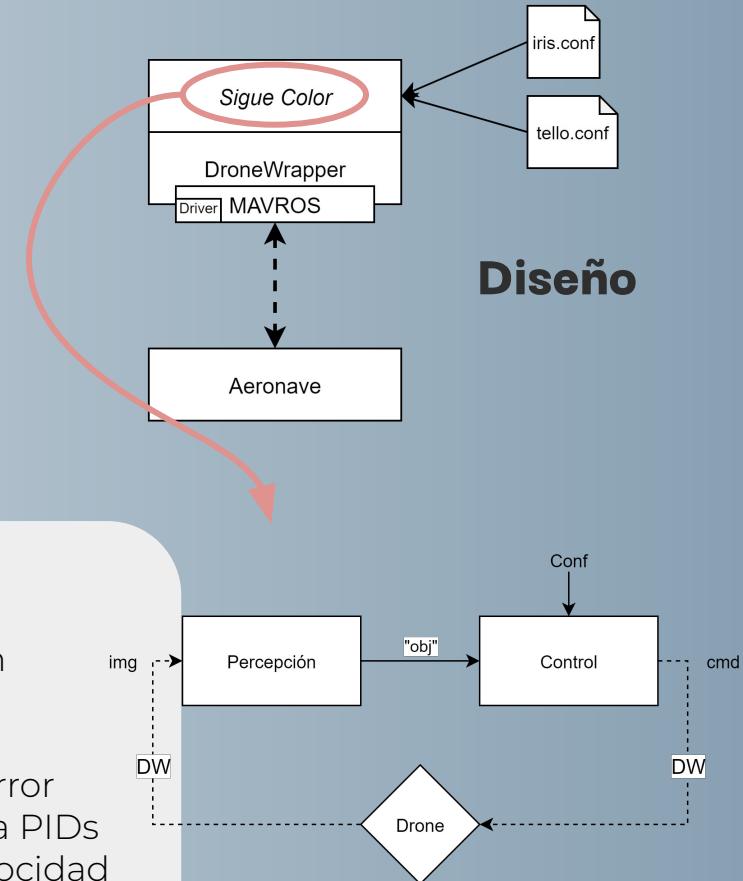


Aplicación

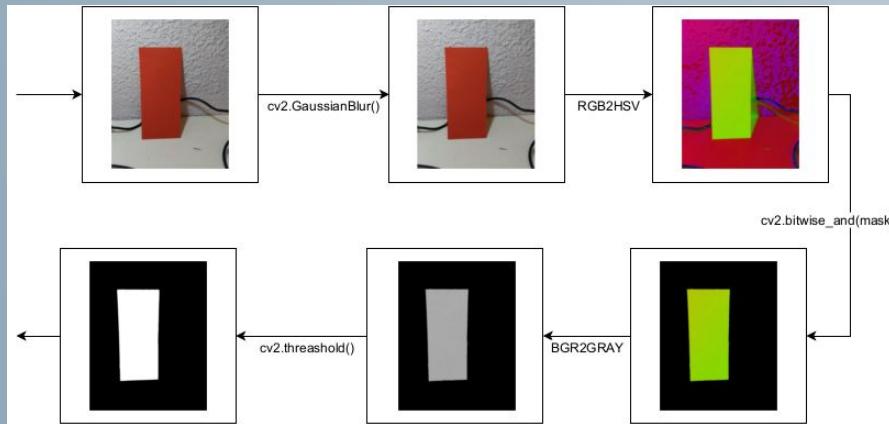
despegue
while:

obten imagen
filtro color
if "color":
nuevo error
actualiza PIDs
cmd velocidad

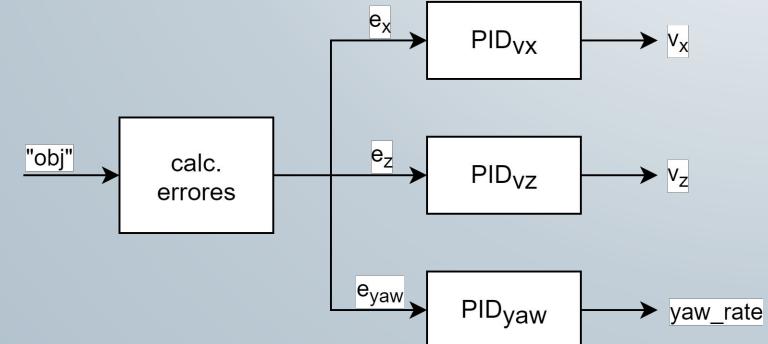
land



Sigue-Color II

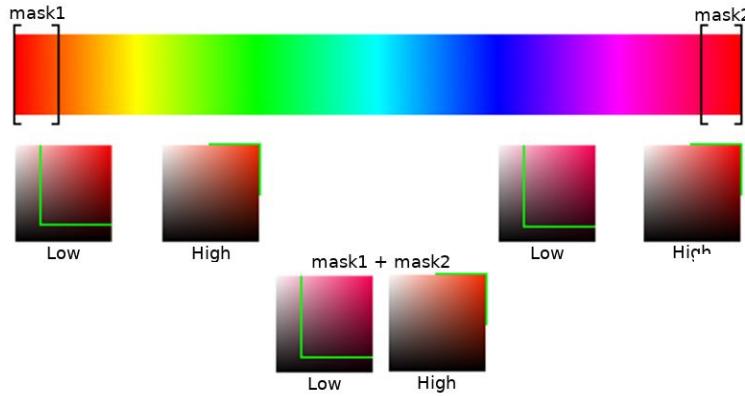


Percepción



Control

Sigue-Color II



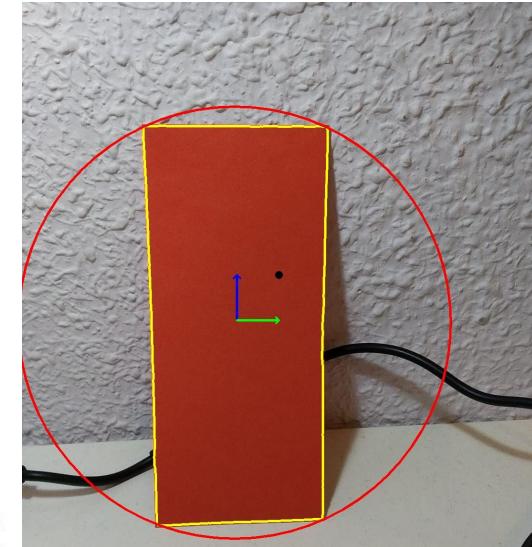
Filtro Color

Máscara	Mínimo			Máximo		
	H	S	V	H	S	V
mask1	0	70	50	10	255	255
mask2	340	70	50	359	255	255

$$e_x = \frac{\text{radio} - \text{target_radio}}{\text{target_radio}}$$

$$e_z = c_y - obj_y$$

$$e_{yaw} = c_x - obj_x$$



Respuesta control

	Iris			Tello		
	kp	ki	kd	kp	ki	kd
vX	0.05	0.0	0.001	0.02	0.0	0.0002
vZ	-0.02	0.0	0.001	-0.0015	0.0	0.0
yaw_rate	-0.005	0.0	0.001	0.002	0.0	0.0001

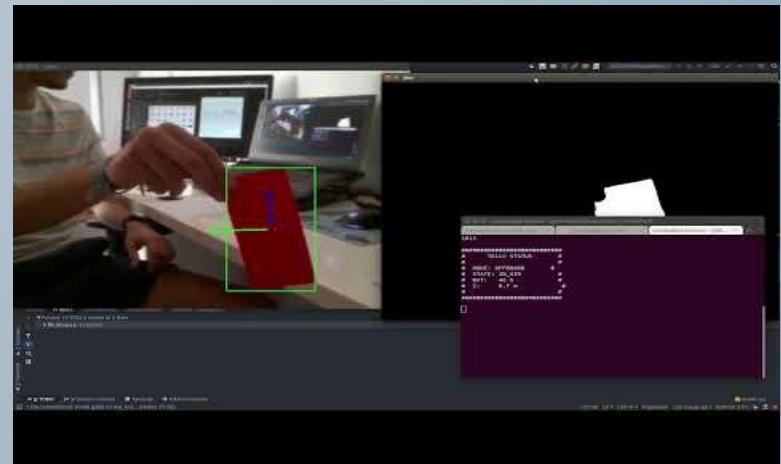
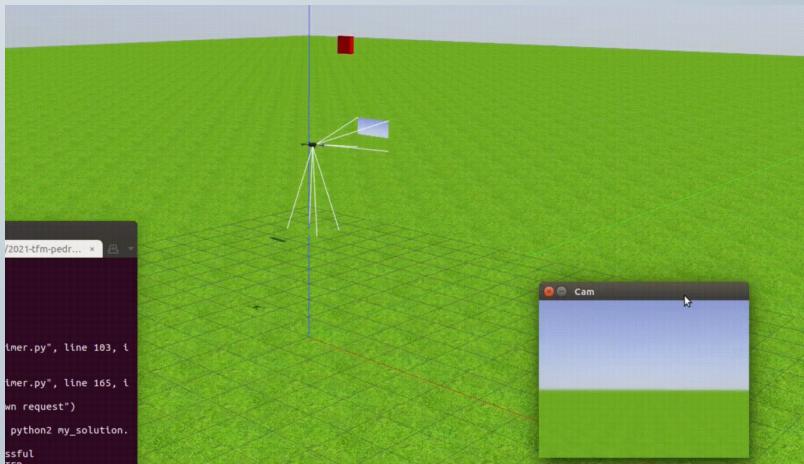
Sigue-Color III



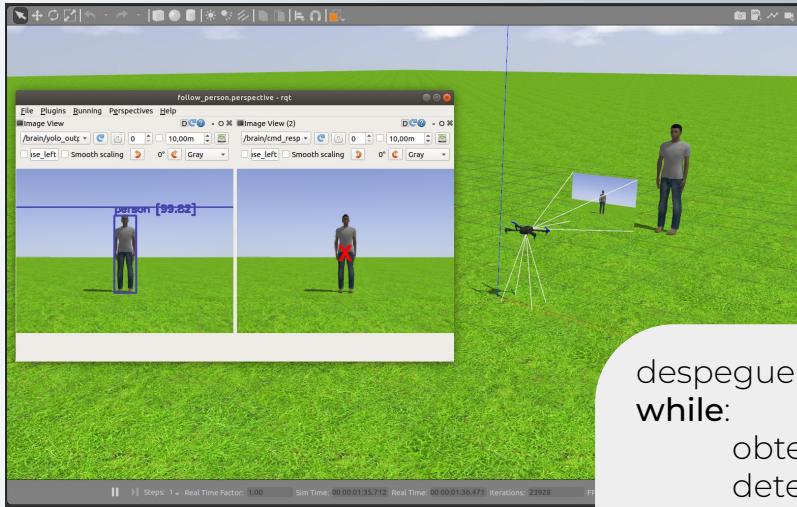
Lista YT

Estático Teleoperado Movimiento

	Estático	Teleoperado	Movimiento
Simulado			
Tello			



Sigue-Persona I



Aplicación

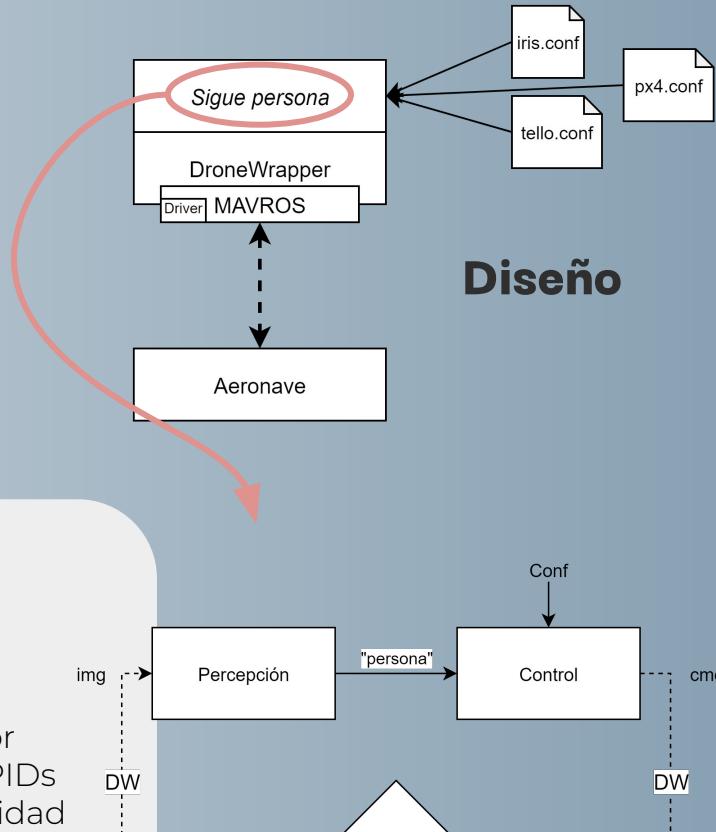
despegue
while:

obten imagen
detecta persona
if "persona":
nuevo error
actualiza PIDs
cmd velocidad

else:

busca persona

land



Sigue-Persona II

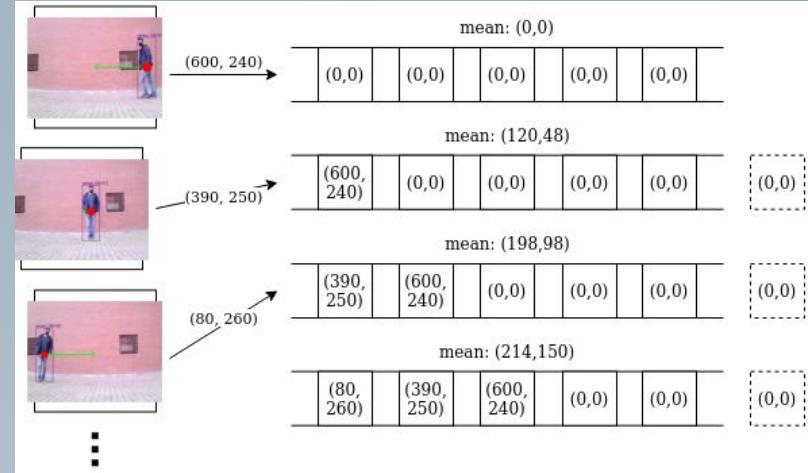
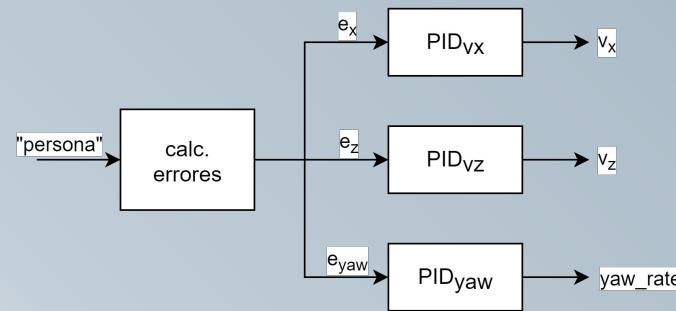
Percepción



Detección



Control



Sigue-Persona II

$$e_x = \frac{area_{total}}{area_{det}} - \frac{area_{total}}{area_{target}}$$

$$e_z = c_y - obj_y$$

$$e_{yaw} = c_x - obj_x$$

	Iris			Tello		
	kp	ki	kd	kp	ki	kd
vx	0.05	0.0	0.001	0.02	0.0	0.0002
vz	-0.02	0.0	0.001	-0.0015	0.0	0.0
yaw_rate	-0.005	0.0	0.001	0.002	0.0	0.0001



Respuesta control

Sigue-Persona III



Lista YT

Estático Movimiento Múltiple

Simulado



Tello

PX4

Sigue-Persona III



Lista YT

Estático

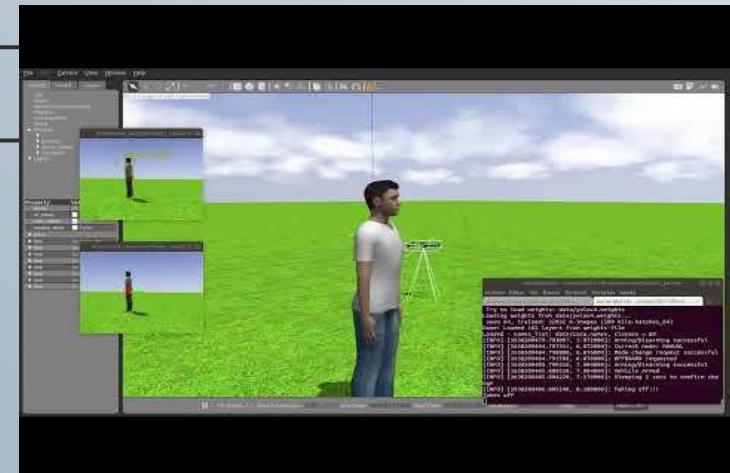
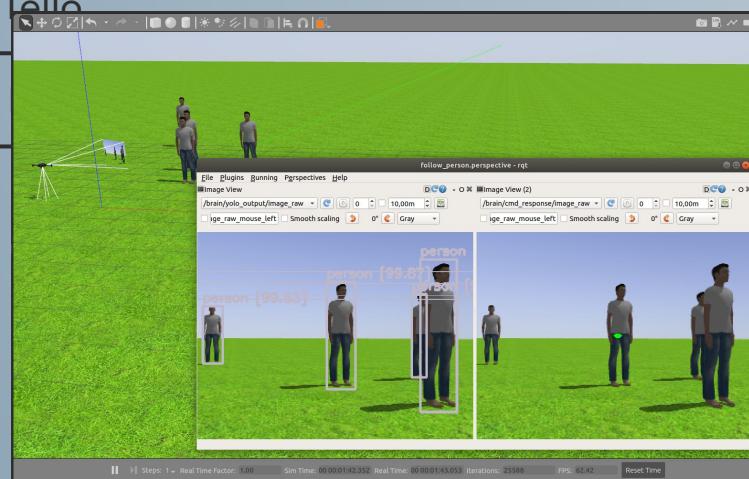
Movimiento

Múltiple

Simulado



Tello



Sigue-Persona III



Lista YT

	Estático	Movimiento	Múltiple
Simulado			
Tello	*	**	***
PX4			

* Maniquí y diferentes personas

** Pasillo y habitación

*** Con y sin occlusiones

Sigue-Persona III



Lista YT

Estático

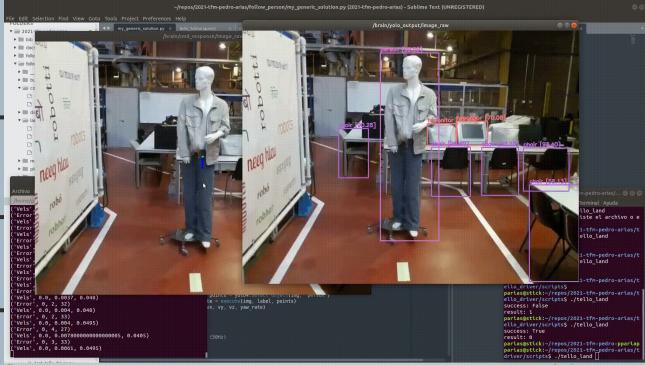
Simulado



Tello



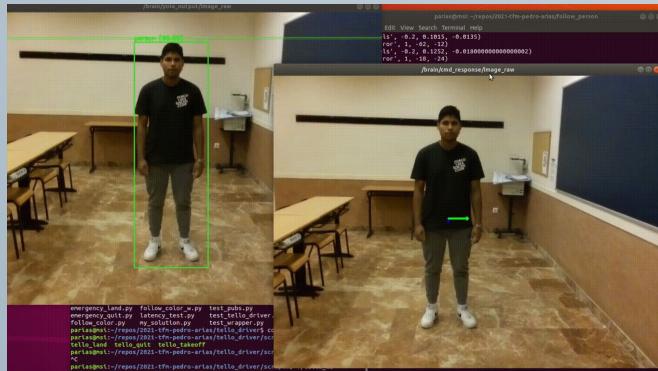
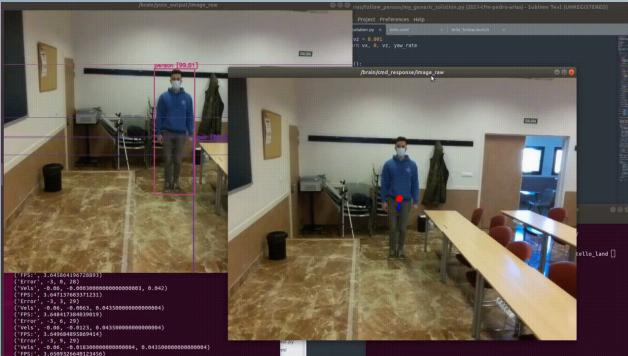
PX4



*Maniquí y diferentes personas

****Pasillo y habitación**

***Con y sin occlusiones



Sigue-Persona III



Lista YT

	Estático	Movimiento	Múltiple
Simulado			
Tello	*	**	***
PX4			***

* Maniquí y diferentes personas

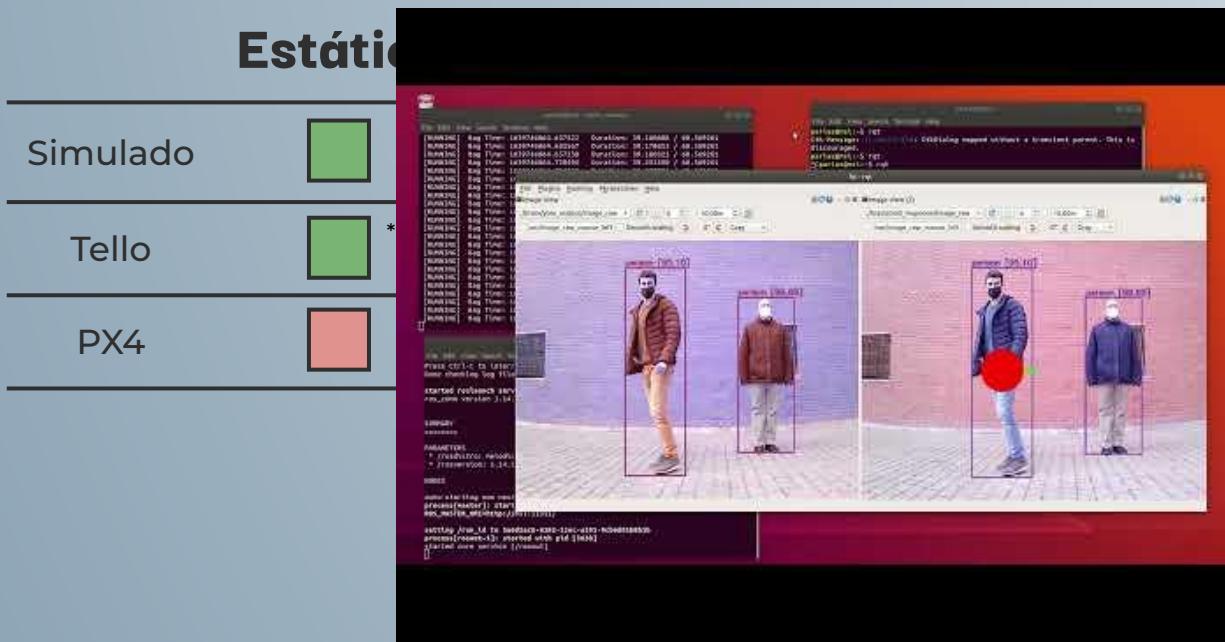
** Pasillo y habitación

*** Con y sin occlusiones

Sigue-Persona III



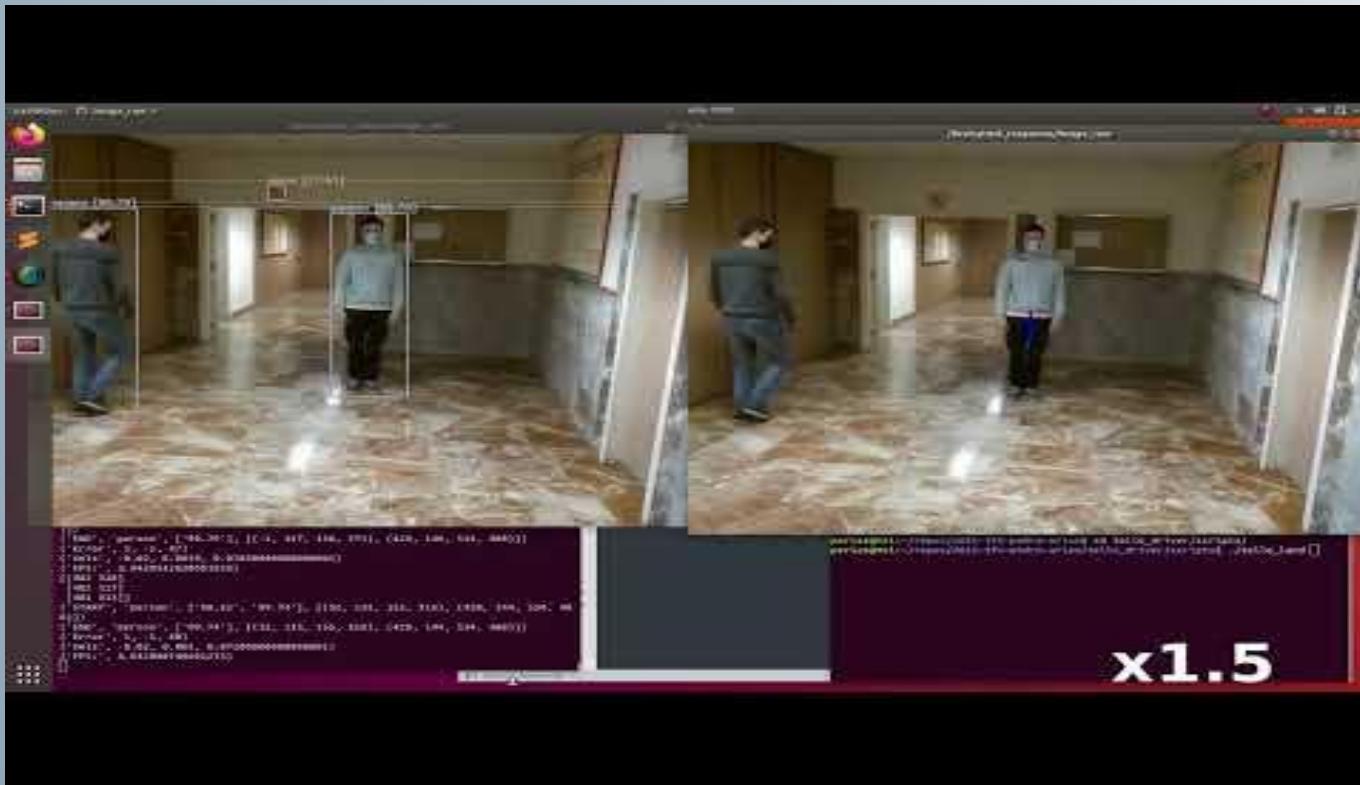
Lista YT



* Maniquí y diferentes personas

** Pasillo y habitación

*** Con y sin occlusiones



Lista YT

4

Conclusiones

Conclusiones
Líneas futuras

Conclusiones



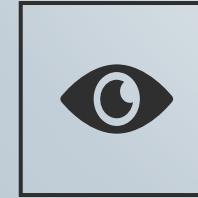
DroneWrapper

Infraestructura operativa utilizada en proyectos de software libre de la comunidad



Plataformas

Tres plataformas de diferente naturaleza probadas



Aplicaciones

Dos aplicaciones de visión de distinta dificultad técnica desarrolladas

Líneas futuras



Nuevas aeronaves

Soporte a nuevas aeronaves



Más funcionalidades

Nuevos sensores: RF, cámara profundidad...

Nuevos métodos API: navegación GPS...



Popularizar su uso

Nuevas aplicaciones

Difusión

Infraestructura de programación de robots aéreos y aplicaciones visuales con aprendizaje profundo

Gracias por su atención

Pedro Arias Pérez

David Martín Gómez
José María Cañas Plaza (URJC)



[JdeRobot/drones](#)

[RoboticsLabURJC/2021](#)
[-tfm-pedro-arias](#)