

# Visual Servoing para el control automático de una torreta

*David Estévez Fernández*  
*Roberto Herrera Esteban*  
*Javier Isabel Hernández*

# Índice

1. Introducción
2. Objetivos
3. Hardware
4. Algoritmo de control
5. Resultados
6. Conclusiones
7. Trabajo futuro

# Introducción



# Motivación

- Implementación de algoritmos de visión en un dispositivo real.
- Ver objetos fuera del campo visual de la cámara
- Posibles aplicaciones en seguridad, ocio, etc.

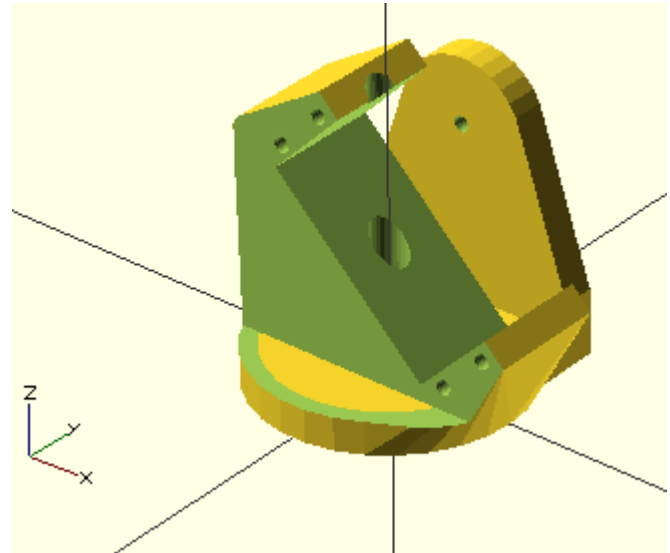


# Objetivos

- Torreta capaz de apuntar a un objetivo de forma autónoma.
- Detección simple de diferentes objetivos.
- Medidas de seguridad: detección de humanos cerca del área de tiro.
- Seleccionar el orden de disparo

# Hardware

- Mecanismo pan-tilt.
- Diseño mecánico e impresión 3D
- Controlador Arduino



# Algoritmo de control

## Algoritmo de control:

- Detección de objetivos
- Tracking (Visual Servoing)
- Filtro de Kalman
- Medidas de seguridad

# Algoritmo de detección (I)

- Segmentación de elementos por color y tamaño.
- Transformación sobre las imágenes para resaltar los colores especificados.

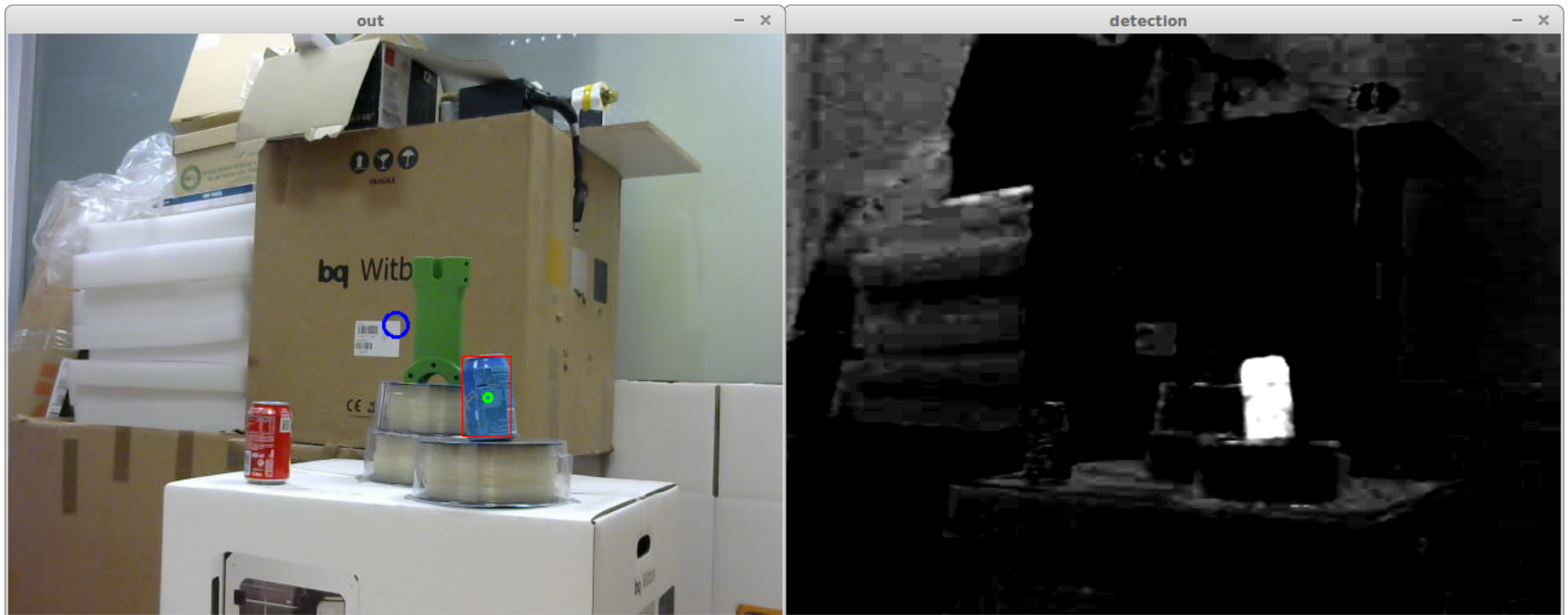




# Algoritmo de detección (II)

Imagen original

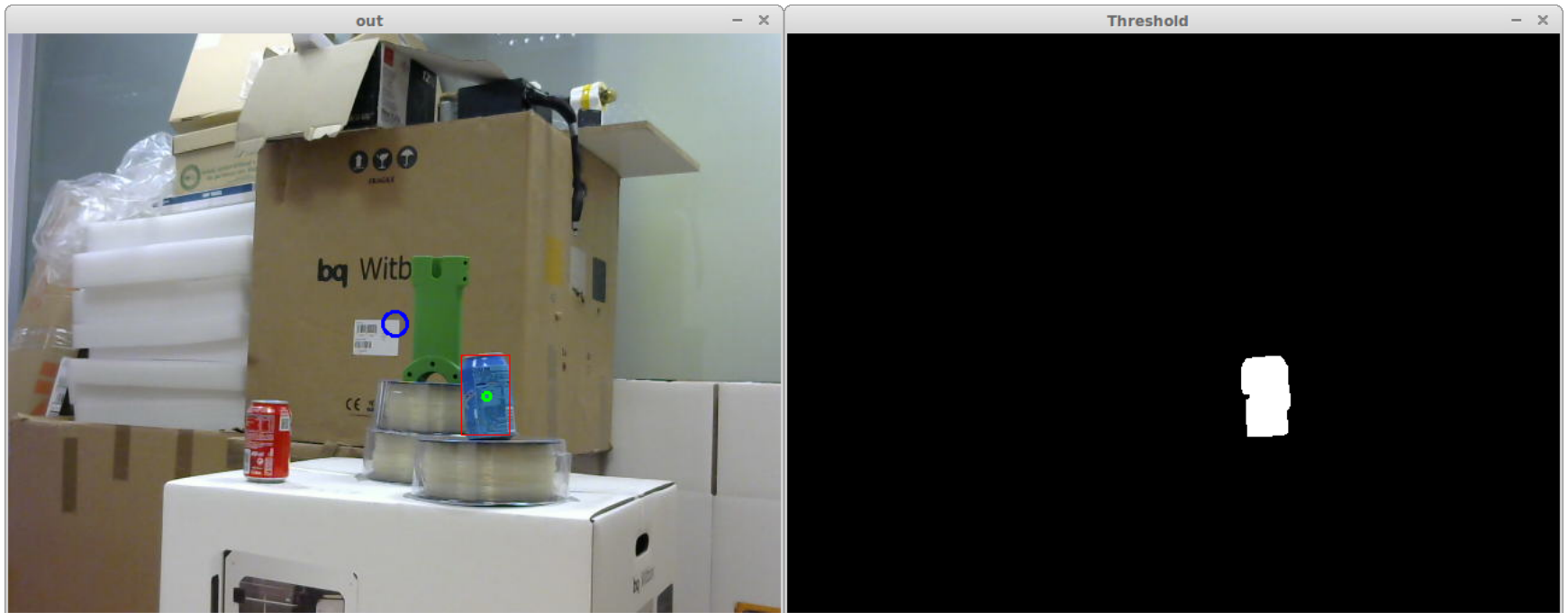
Imagen transformada



# Algoritmo de detección (III)

Imagen original

Imagen umbralizada



# Visual Servoing

- Consiste en cerrar el lazo de control en posición mediante visión.
- Suele tener matemáticas muy complejas (jacobiana visual) así que ha sido simplificado para este trabajo.

# Visual Servoing



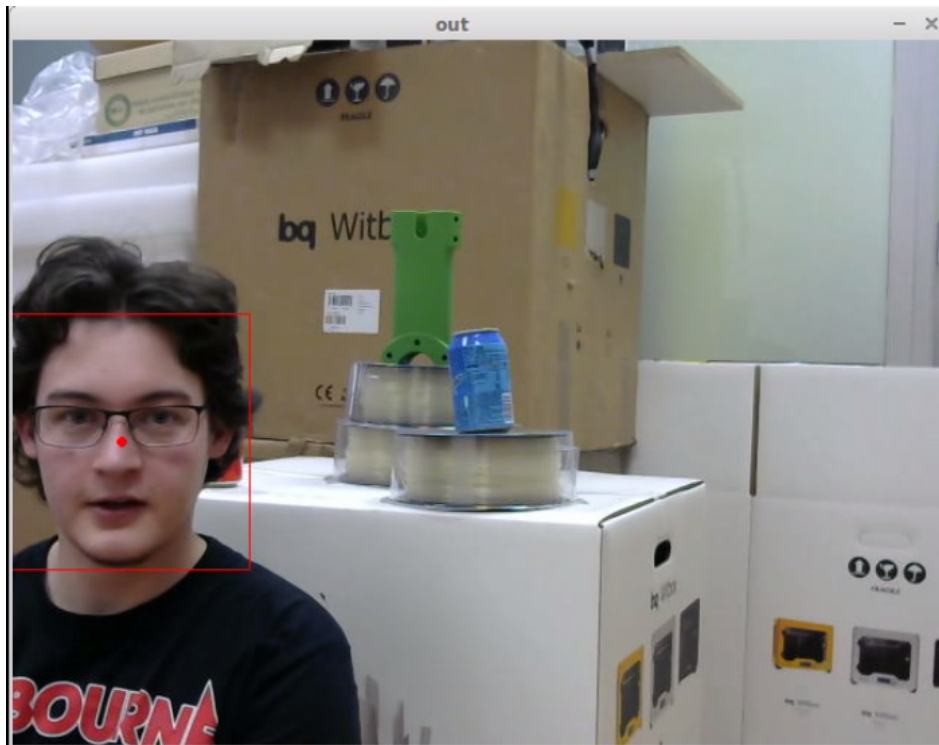
- Se calcula el error en posición mediante visión, y se controlan los motores con un control proporcional.

# Filtro de Kalman

- Es un estimador del estado de los blancos
- Filtra el ruido en la posición del centroide
- Filtra el ruido ocasionado por la vibración y movimiento de la torreta

# Medidas de seguridad

- Detección de caras mediante Cascadas de Haar (**Haar Feature-based Cascade Classifiers**)



# Resultados



# Conclusiones

- Se han ampliado los conocimientos sobre OpenCV.
- Se ha desarrollado un HW para emular una torreta con Pan-Tilt.
- Las imágenes en RGB han dado mejor resultado que las HSV en esta aplicación.
- Se ha cumplido los objetivos propuestos.



# Trabajos futuros (I)

- Mejorar el Tracking para predecir la posición del objeto según el movimiento de la torreta.
- Objetivos móviles.
- Buscar los objetivos que no estén dentro del rango de visión de la cámara pero sí de la torreta.

# Trabajos futuros (II)

- Robot Devastation (Asrob).
- Crear una interfaz gráfica.

# Visual Servoing para el control automático de una torreta

¡Muchas  
gracias!

*David Estévez Fernández  
Roberto Herrera Esteban  
Javier Isabel Hernández*