
Técnicas de autocalización en 3D con visión monocular

Eduardo Perdices

eperdices@gsyc.es



Universidad Rey Juan Carlos
05 Abril 2016

Contenidos

- Introducción
- MonoSLAM
- PTAM
- SVO
- Futuro Próximo

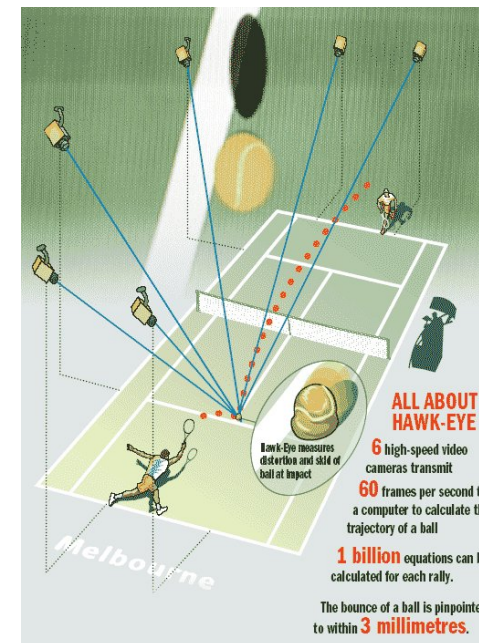
Introducción

- Visión Artificial: campo de la informática que estudia el uso de cámaras
- Bajo coste
- Permite extraer mucha información
- Procesamiento muy costoso
- Detección de bordes, features, texturas...
- Reconocimiento de formas, caras...



Aplicaciones

- Medicina
- OCR
- Biometría
- Ojo de halcón (tenis)
- MediaPro: distancia recorrida por un futbolista



Autocalización con cámaras

- Visión monocular
- Cámaras estéreo: Bumblebee
- Cámaras RGBD: Kinect, Xtion, Kinect2
- Requisitos: tiempo real, robustez



Aplicaciones de la autocalización

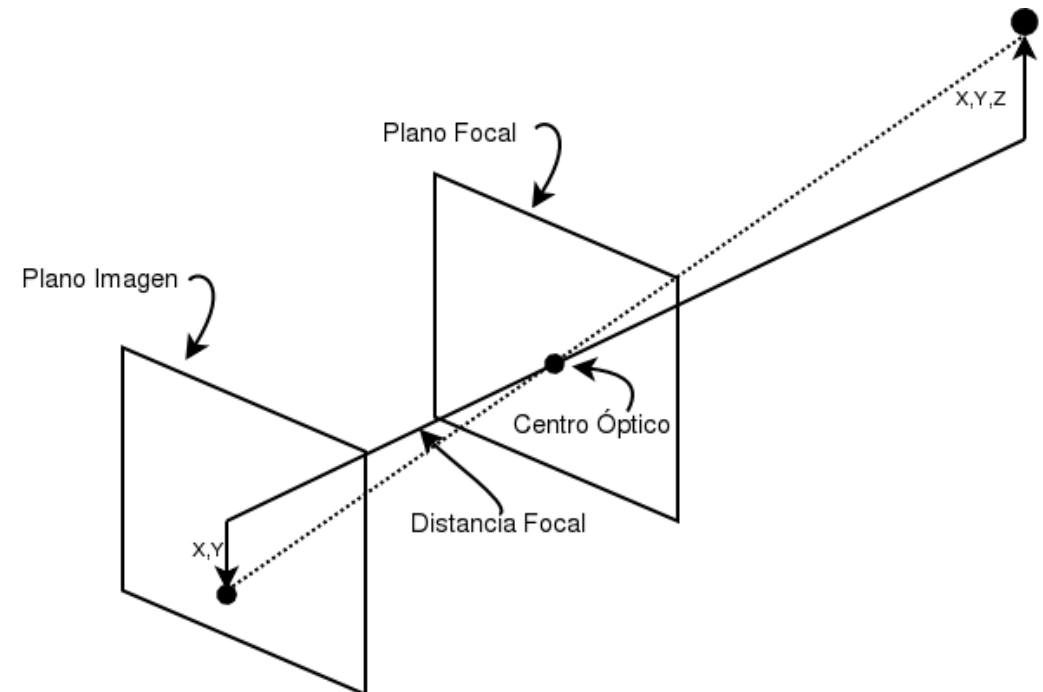
- Navegación
- Mapas, reconstrucción 3D
- Realidad aumentada
- Transporte
- Vigilancia
- Vehículos autónomos



Extracción de información

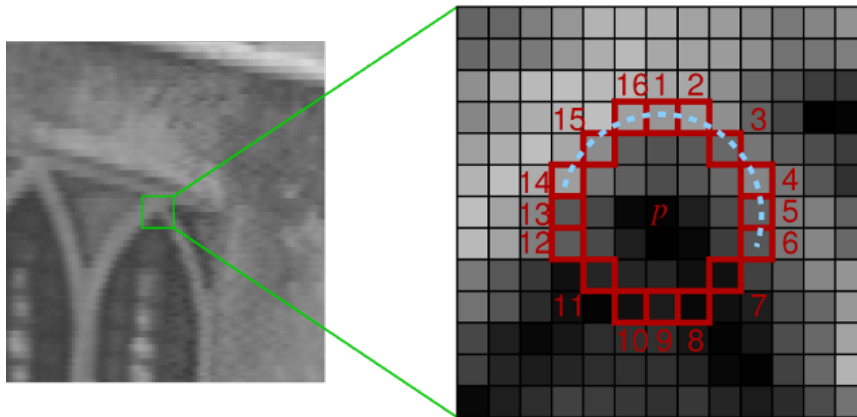
Modelo Pin-Hole

- Parámetros intrínsecos
 - Distancia focal
 - Centro óptico
 - Skew (distorsión...)
- Parámetros extrínsecos: posición y orientación
- Permite sacar información espacial



Detección de puntos

- Puntos con rica información local a su alrededor
- Deben ser repetibles e invariantes a escala y orientación
- Múltiples algoritmos de detección (SIFT, SURF, FAST, ORB)
- FAST: Destaca por su rapidez, muy utilizado para tiempo real



Matching

- Utilizando descriptores (SIFT, SURF, ORB)
- Mediante comparación de parches:
 - Diferencia de píxeles
 - Color medio, conteo de colores, etc
 - Suma de diferencias cuadráticas (SSD, ZSSD, etc)

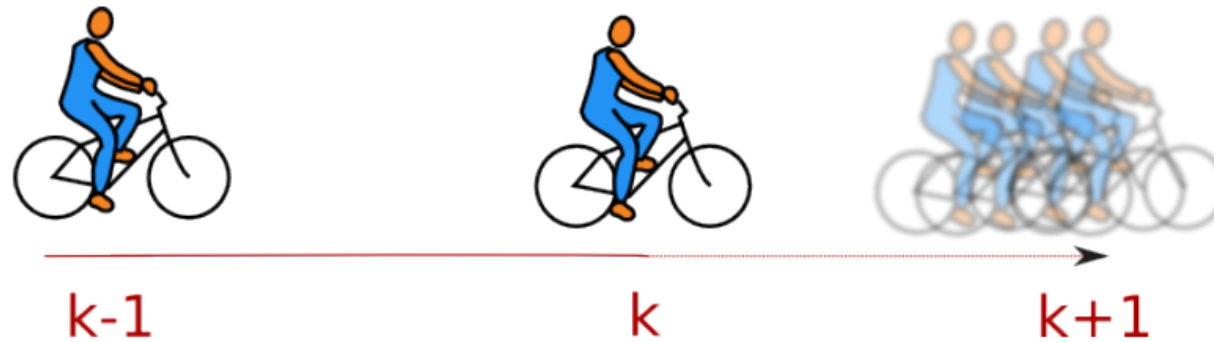


MonoSLAM

- Siglas de Simultaneous Localization and Mapping
- Objetivos:
 - Localización precisa de robots/cámaras en 3D
 - Generación dinámica de mapas en 3D
 - Funcionamiento en tiempo real (30FPS)
- Manejo de hasta 30-100 puntos por iteración

Filtro de Kalman

- Estimación de estados basado en observaciones
- Sistemas lineales (KF) o no lineales (EKF)



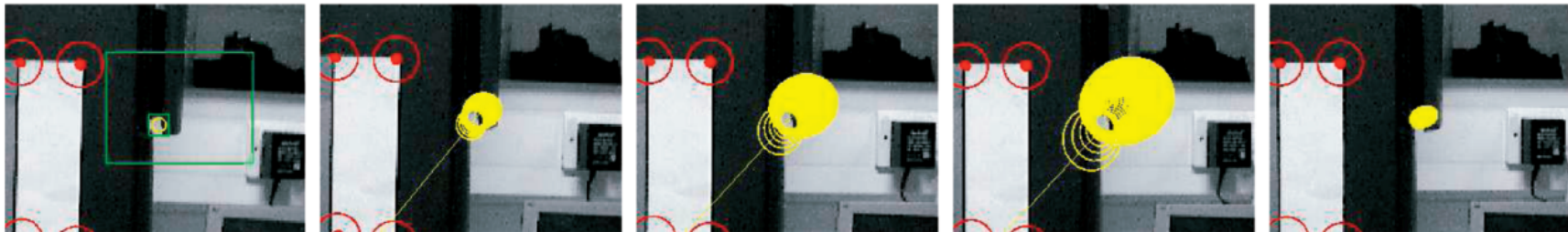
- Vector de estados en MonoSLAM:
 - Cámara: Posición, orientación, velocidad lineal y angular
 - Para cada punto: Posición en 3D

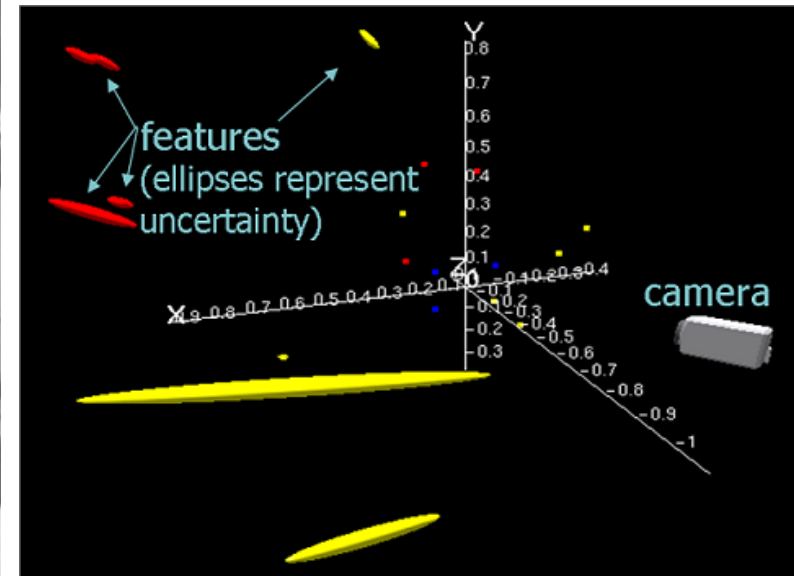
Primera aproximación: Partiendo de puntos conocidos

- Arranque con al menos 4 puntos conocidos en 3D
- Detección de puntos con FAST
- Matching entre parches (11x11)
- Área de búsqueda depende de incertidumbre

Primera aproximación: Inicialización de puntos

- Se necesita desplazamiento de la cámara y varias observaciones
- Inicialización de una línea en 3D con profundidad desconocida
- Sólo se inicializan puntos “cercanos” a los actuales
- Distribución de N partículas a lo largo de la línea en 3D
- Se inserta el punto en el EKF sólo cuando haya poca incertidumbre





Puntos en el infinito: Inverse Depth Parametrization

- No necesita partir de puntos conocidos
- Pueden inicializarse puntos cercanos o lejanos (infinito)
- Puntos dentro del EKF desde su inicialización (mejora orientación)
- Problema: El mapa generado depende de la escala

$$\mathbf{x}_i = \begin{pmatrix} X_i \\ Y_i \\ Z_i \end{pmatrix} = \begin{pmatrix} x_i \\ y_i \\ z_i \end{pmatrix} + \frac{1}{\rho_i} \mathbf{m}(\theta_i, \phi_i)$$

$$\mathbf{m} = (\cos \phi_i \sin \theta_i, -\sin \phi_i, \cos \phi_i \cos \theta_i)^\top$$

Otras técnicas

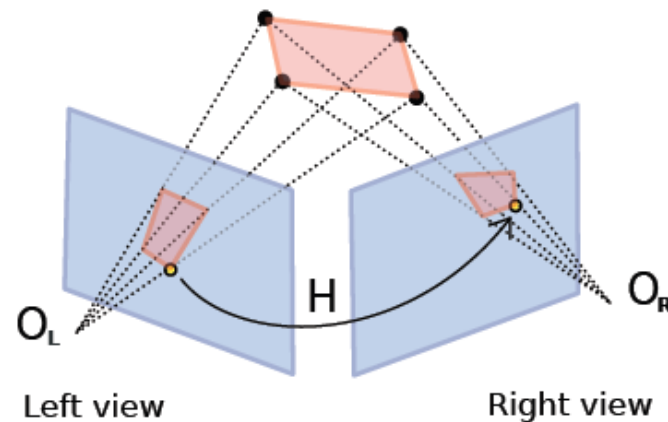
- One-point RANSAC
- Active matching
- Búsqueda de estructuras de puntos
- Puntos dinámicos

PTAM

- Siglas de Parallel Tracking and Mapping
- División del algoritmo en dos hilos (tracking y mapping)
- Tracking en tiempo real (30FPS)
- Mapas optimizados mediante Bundle Adjustment
- Cálculo de hasta 1000 puntos por iteración
- Permite relocalización

Inicialización por Homografía

- Homografía: Relaciona pares de puntos coplanares entre dos imágenes
- Permite obtener rotación y traslación a partir de H
- Obtención de puntos 3D por triangulación (dependientes de escala)
- Selección de puntos coplanares utilizando RANSAC



$$x' = Hx$$

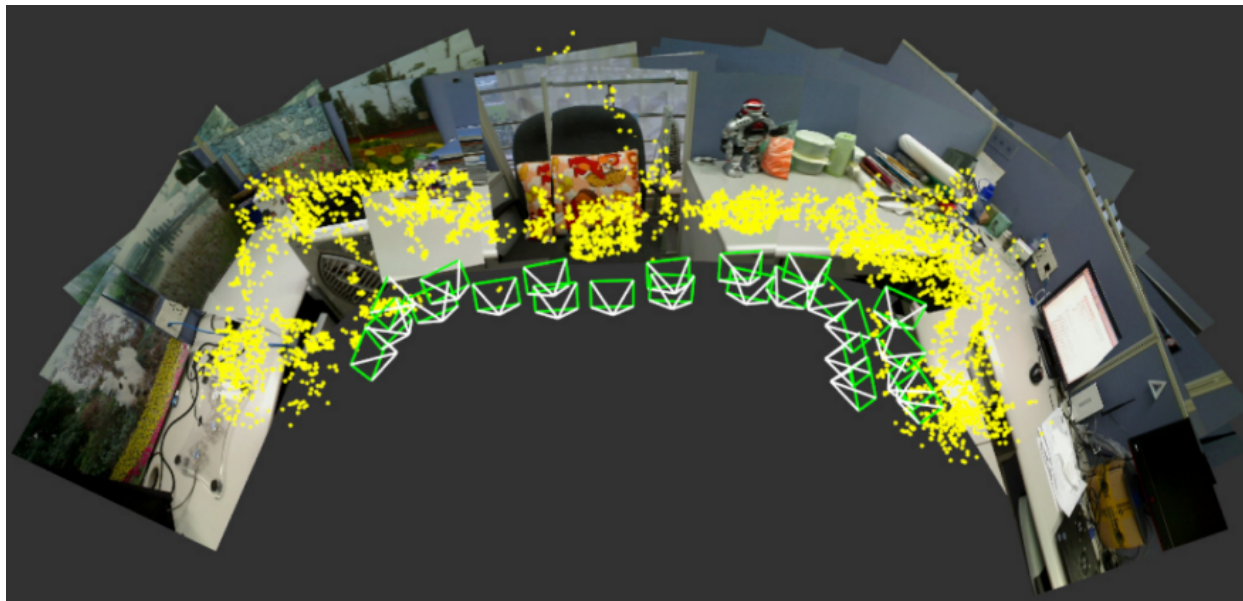
$$\begin{bmatrix} x'_x \\ x'_y \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x_x \\ x_y \\ 1 \end{bmatrix}$$

Tracking

- Calcula la posición, orientación y velocidades de la cámara
- Detección de puntos con FAST y matching entre parches (8x8)
- Área de búsqueda fijo
- Pasos del algoritmo:
 - Predicción de posición actual con modelo de movimiento
 - Búsqueda de grano grueso (50 puntos) y grano fino (1000 puntos)
 - Actualización de posición mediante Gauss-Newton
- Comprueba la calidad del tracking y si necesita relocalizarse

Creación de nuevos puntos

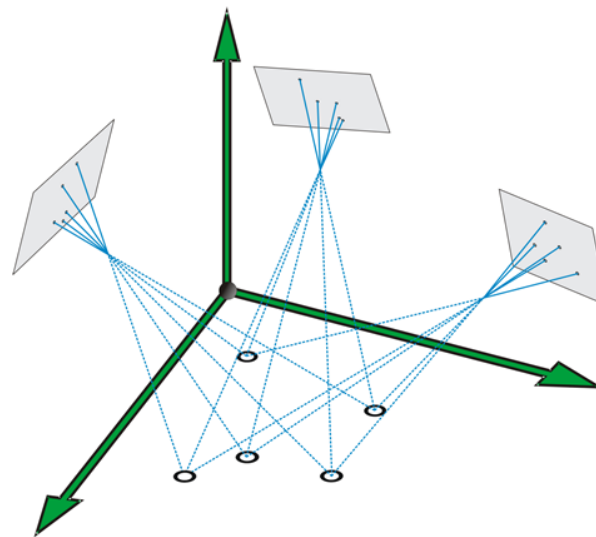
- Se guardan determinados frames en memoria (KeyFrames)
- Se obtienen nuevos puntos triangulando entre KeyFrames



Mapping

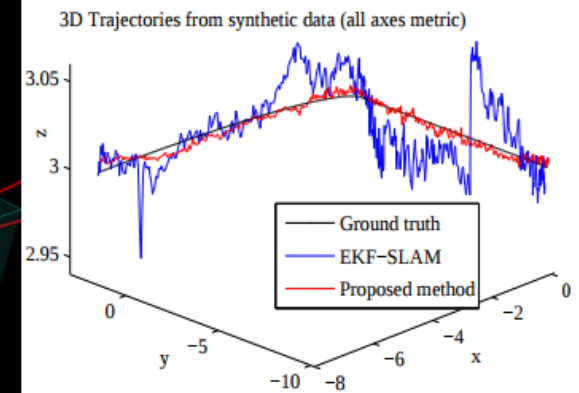
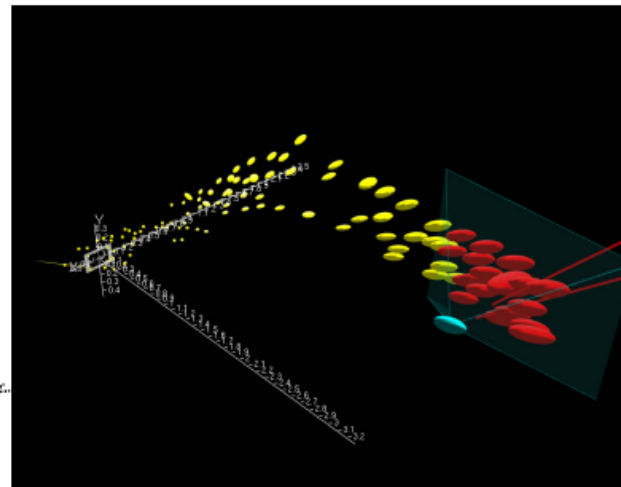
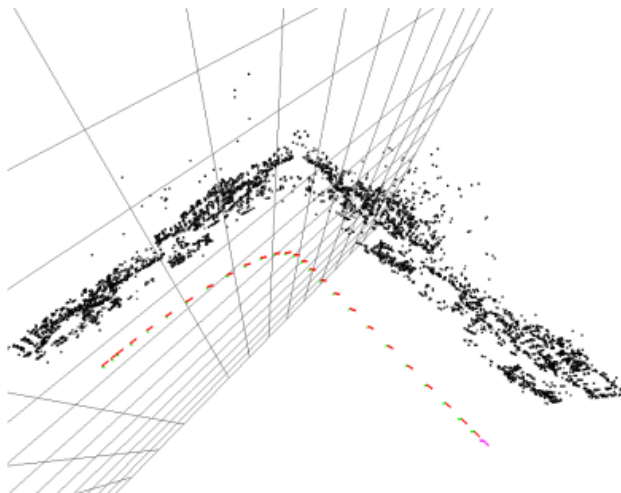
- Almacena puntos 3D (miles) y KeyFrames (cientos)
- No necesita funcionar en tiempo real
- Optimización de posiciones mediante Bundle Adjustment:
 - Minimiza el error de reproyección entre puntos y frames

$$\min_{\mathbf{a}_j, \mathbf{b}_i} \sum_{i=1}^n \sum_{j=1}^m v_{ij} d(\mathbf{Q}(\mathbf{a}_j, \mathbf{b}_i), \mathbf{x}_{ij})^2$$



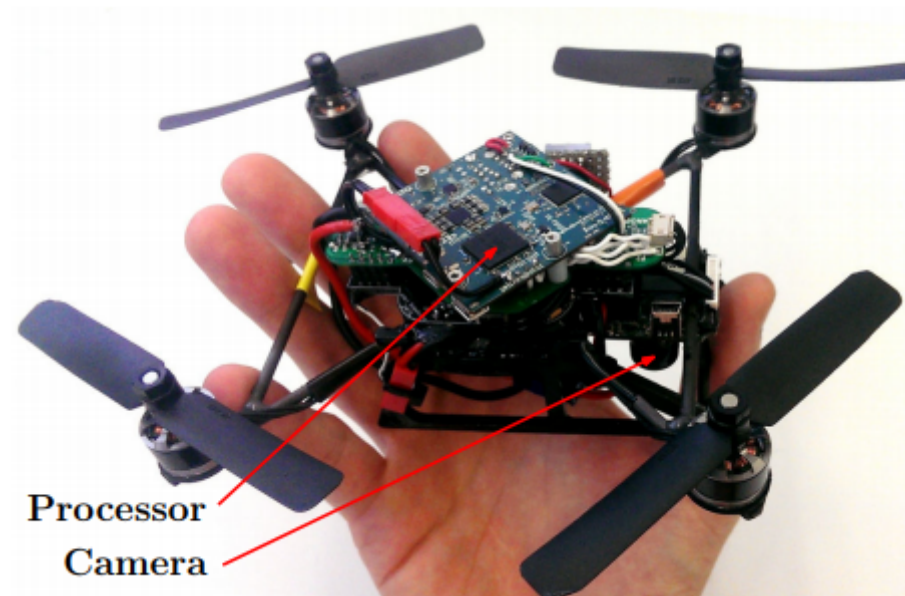
Mejoras de PTAM vs MonoSLAM

- Maneja más puntos (~ 100 veces más)
- La localización es más precisa
- El mapa generado es más realista
- Relocalización



SVO

- Siglas de Semi-Direct Monocular Visual Odometry
- Objetivo: Tiempo real usando procesadores poco potentes

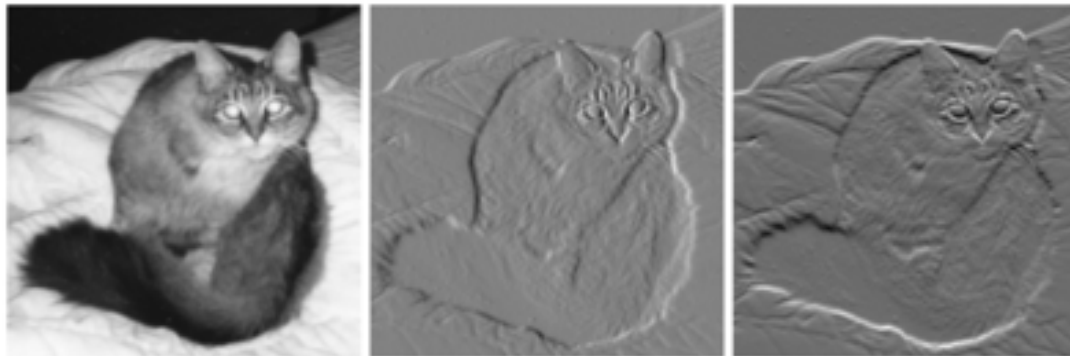


Características

- Similitudes con PTAM:
 - División del algoritmo en dos hilos (tracking y mapping)
 - Inicialización mediante homografía
 - Permite relocalización
- Diferencias con PTAM:
 - Tracking con métodos directos
 - Creación de puntos mediante profundidad inversa
- Tracking mucho más eficiente ($\sim 100\text{FPS}$)
- Autocalización más precisa

Tracking

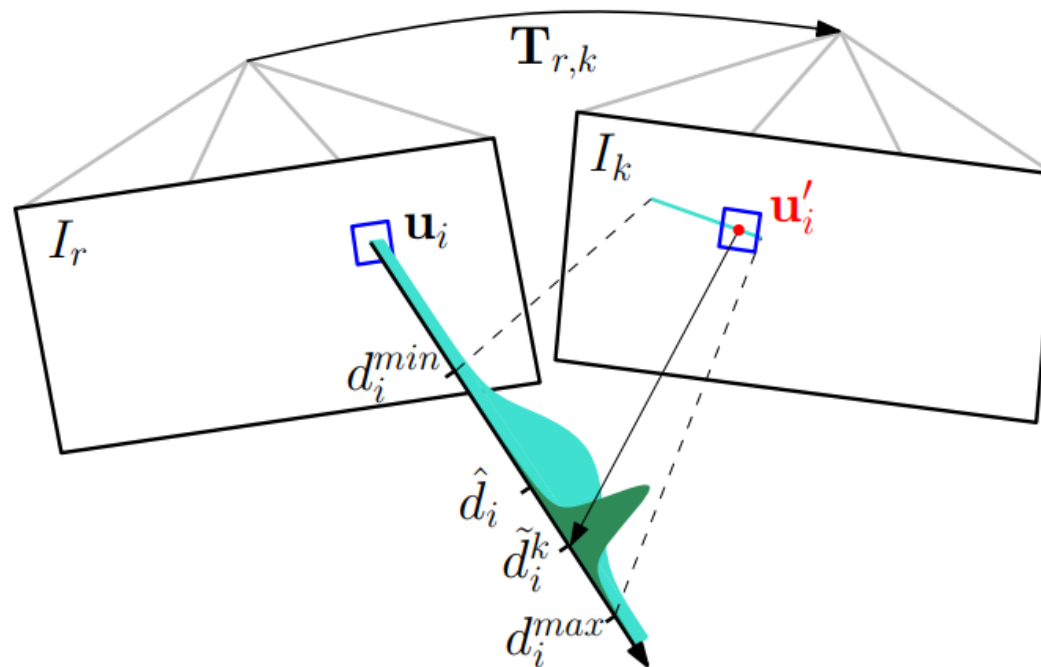
- Método directo para estimar desplazamiento:
 - Minimiza la diferencia de intensidad entre dos imágenes
 - Uso de imagen completa o de puntos destacados
 - Ideal para imágenes con poca textura o borrosas



- Aumento de precisión con matching de puntos:
 - Pocos puntos (120-150) distribuidos en la imagen
 - Regiones muy pequeñas

Inicialización de puntos

- Puntos en el infinito: Inverse Depth Parametrization
- Profundidad actualizada fusionando densidades de probabilidad
- Sólo se añaden al mapa los puntos que convergen

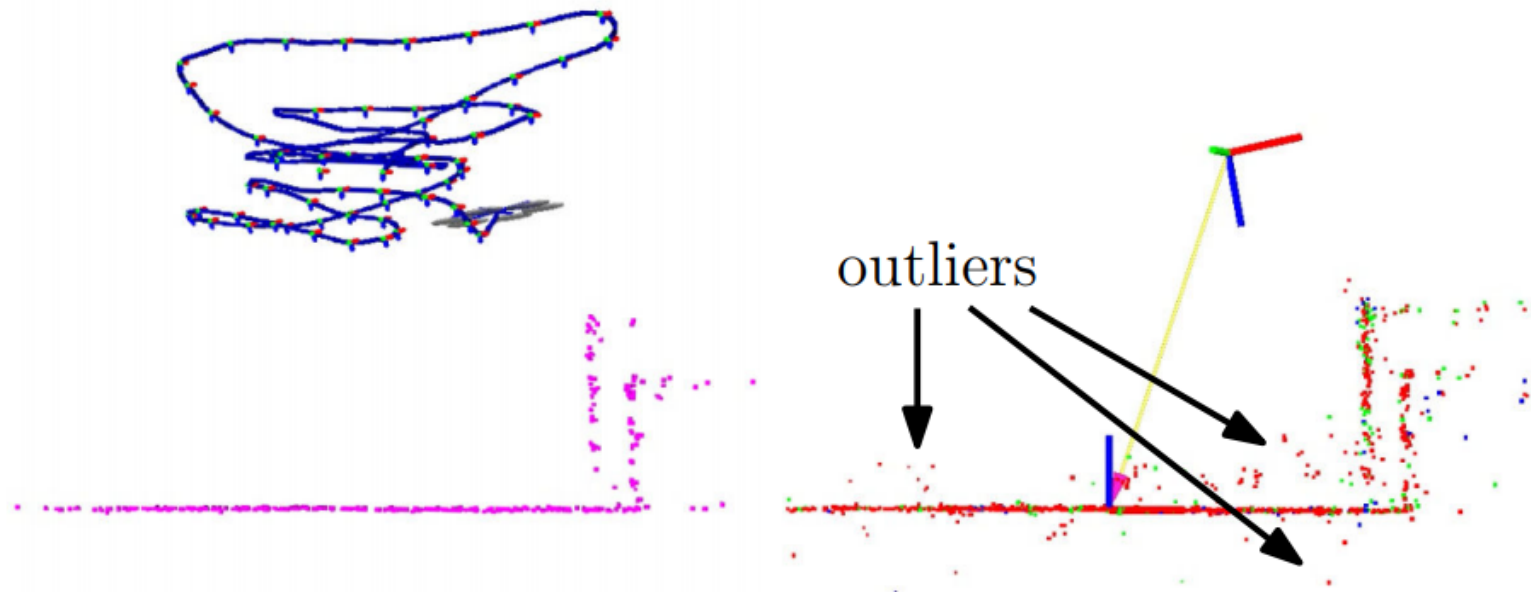


Mapping

- Funcionamiento similar a PTAM
- Guarda puntos 3D y KeyFrames
- Pocos outliers
- Mapa menos denso pero más fiable
- Posibilidad de utilizar Bundle Adjustment

Mejoras de SVO vs PTAM

- Menor coste computacional
- Mayor precisión
- Puntos más fiables (pero menos puntos)
- Funciona en entornos con poca textura



Futuro próximo

- Realidad aumentada en móviles
- Modelado 3D
- Aplicaciones médicas
- Vehículos autónomos
- Hologramas





Técnicas de autocalización en 3D con visión monocular

Eduardo Perdices

eperdices@gsyc.es



Universidad Rey Juan Carlos
05 Abril 2016
