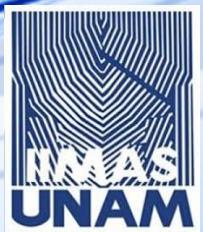


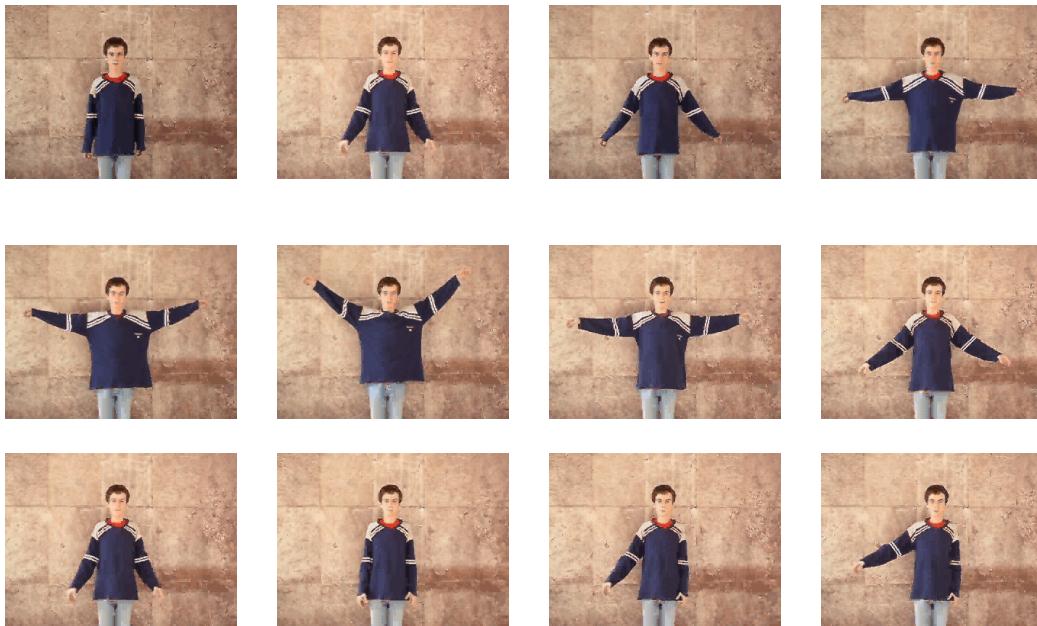
Estimación de Movimiento



Dr. Jorge Perez-Gonzalez



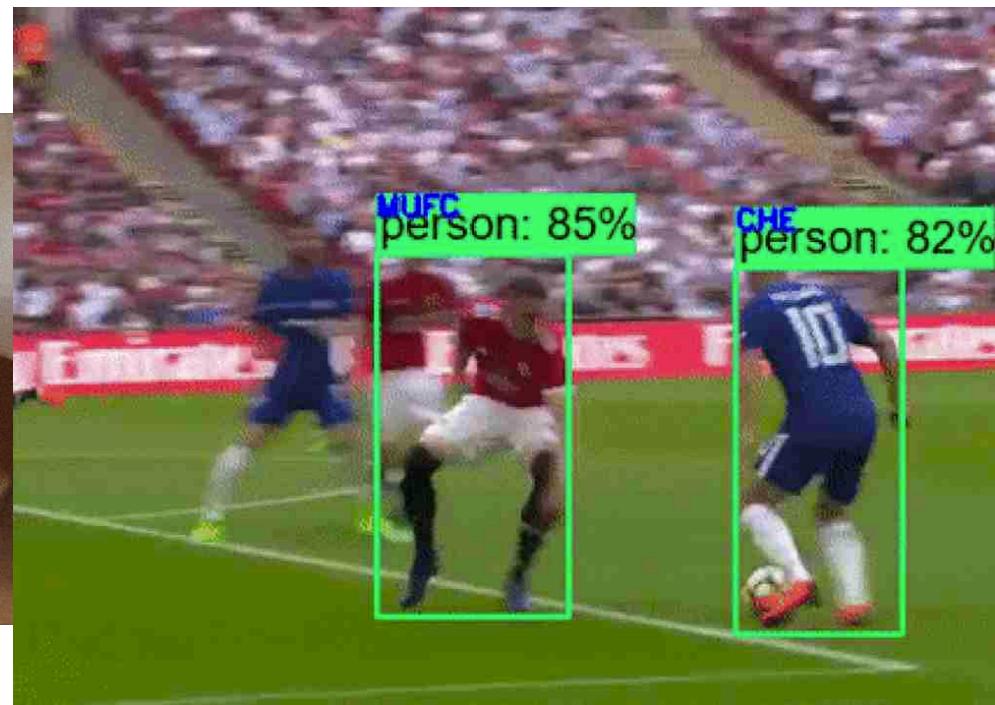
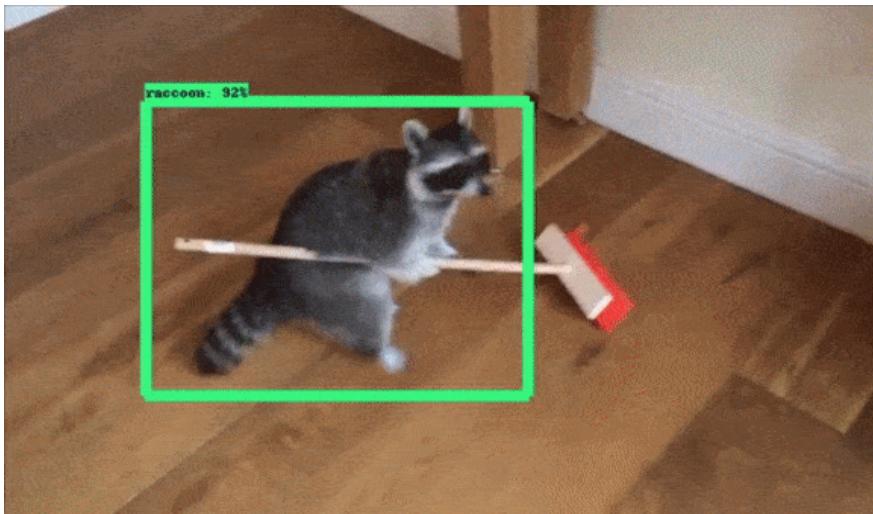
Problema



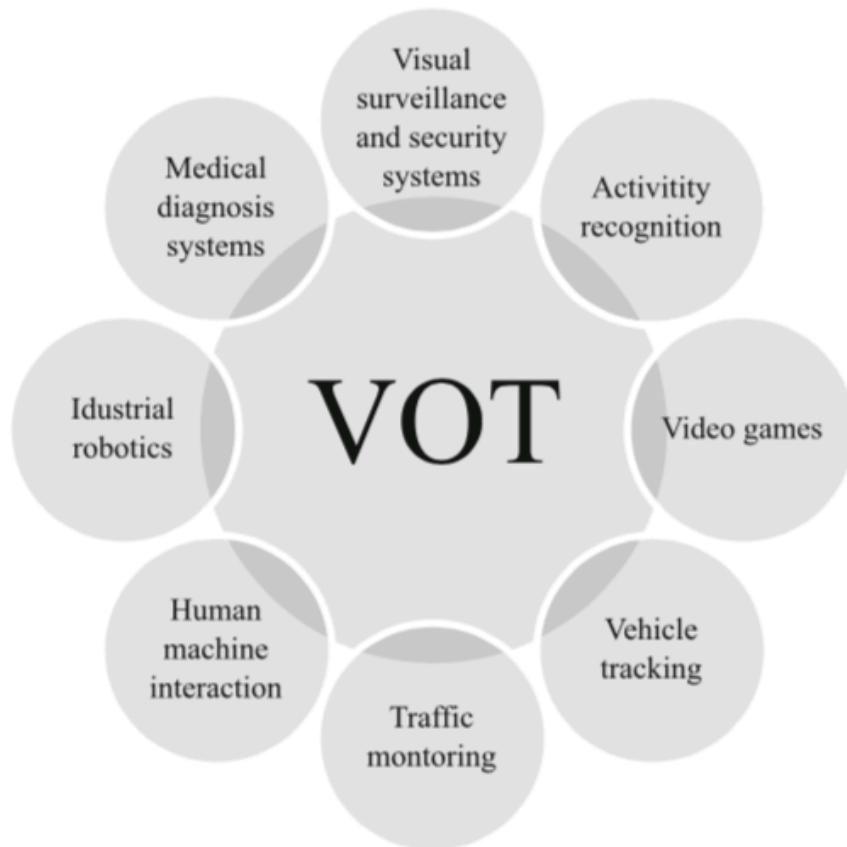
Dr. Jorge Perez-Gonzalez



Aplicaciones



Visual object tracking (VOT)



Algoritmos clasicos

- Diferencias entre imágenes
- Substracción del fondo
- Estimación de bloques de movimiento
- Flujo Óptico

Diferencias entre imágenes



Detect changes between two images

Input $I_t[r, c]$ and $I_{t-\delta}[r, c]$: two monochrome input images taken δ seconds apart.

Input τ is an intensity threshold.

$I_{out}[r, c]$ is the binary output image; \mathbf{B} is a set of bounding boxes.

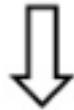
1. For all pixels $[r, c]$ in the input images,
set $I_{out}[r, c] = 1$ if ($|I_t[r, c] - I_{t-\delta}[r, c]| > \tau$)
set $I_{out}[r, c] = 0$ otherwise.
2. Perform connected components extraction on I_{out} .
3. Remove small regions assuming they are noise.
4. Perform a closing of I_{out} using a small disk to fuse neighboring regions.
5. Compute the bounding boxes of all remaining regions of changed pixels.
6. Return $I_{out}[r, c]$ and the bounding boxes \mathbf{B} of regions of changed pixels.

Algorithm 16: Detection of change via image subtraction

Substracción del fondo

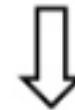
Image at time t:

$$I(x, y, t)$$



Background at time t:

$$B(x, y, t)$$



$$> Th$$

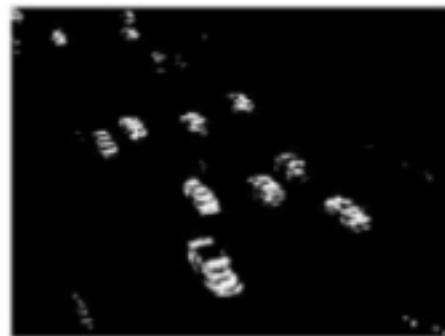
1. Estimate the background for time t.
2. Subtract the estimated background from the input frame.
3. Apply a threshold, Th , to the absolute difference to get the foreground mask.

Resultados

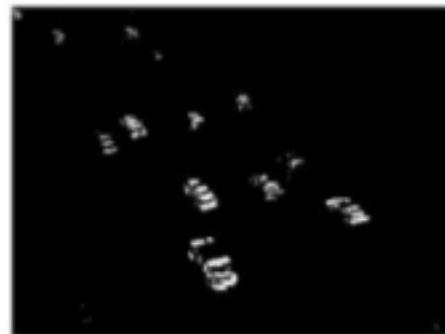
$\text{Th} = 25$



$\text{Th} = 50$



$\text{Th} = 100$



$\text{Th} = 200$



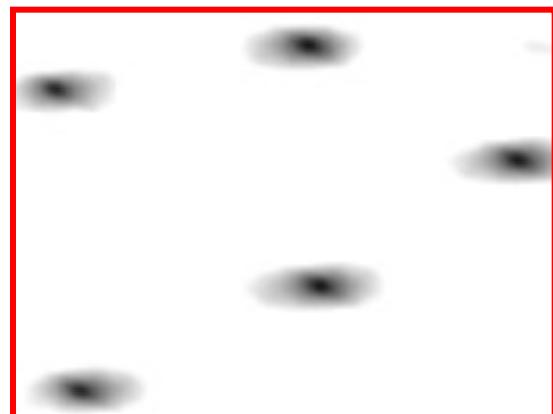
Imagen de entrada A (239x156)

P - patrón a
buscar (68x37)



Mapa de
matching

M



Mapa superpuesto

$$M(x, y) := \sum_{a=0..w-1} \sum_{b=0..h-1} (P'(a, b) \cdot A'(x+a, y+b))$$

Esto es lo que se llama la correlación

donde $P'(a,b) := P(a,b) - \text{Media}(P)$. Lo mismo para A' .

- El valor (normalizado) está entre -1 y +1. Cuanto mayor (más próximo a +1) más probabilidad.

Imagen de entrada, **A**

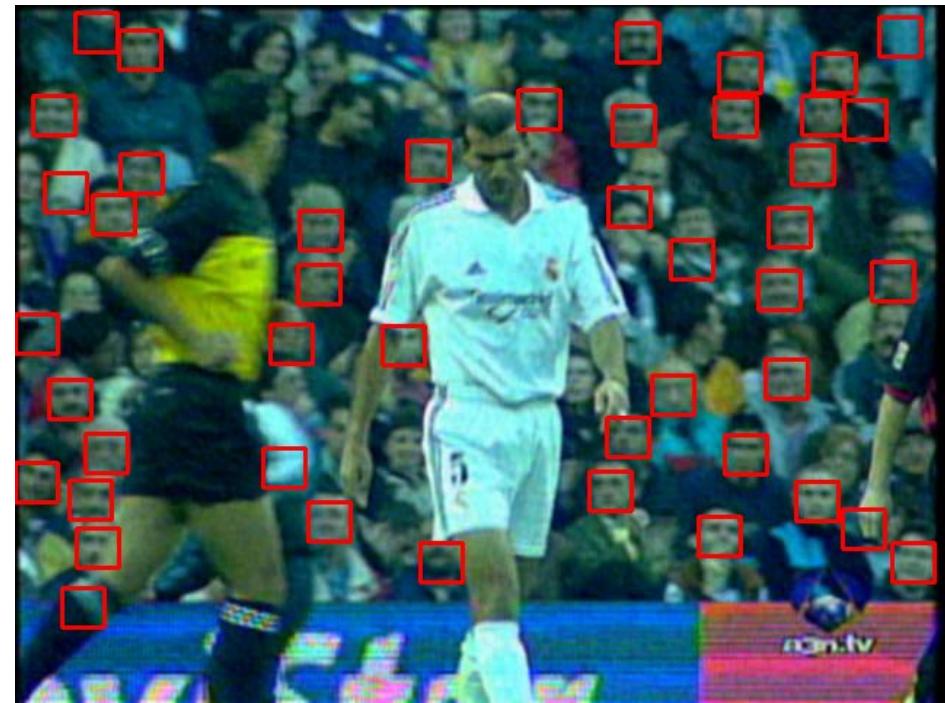
Patrón, **P**



Mapa de matching, **M**



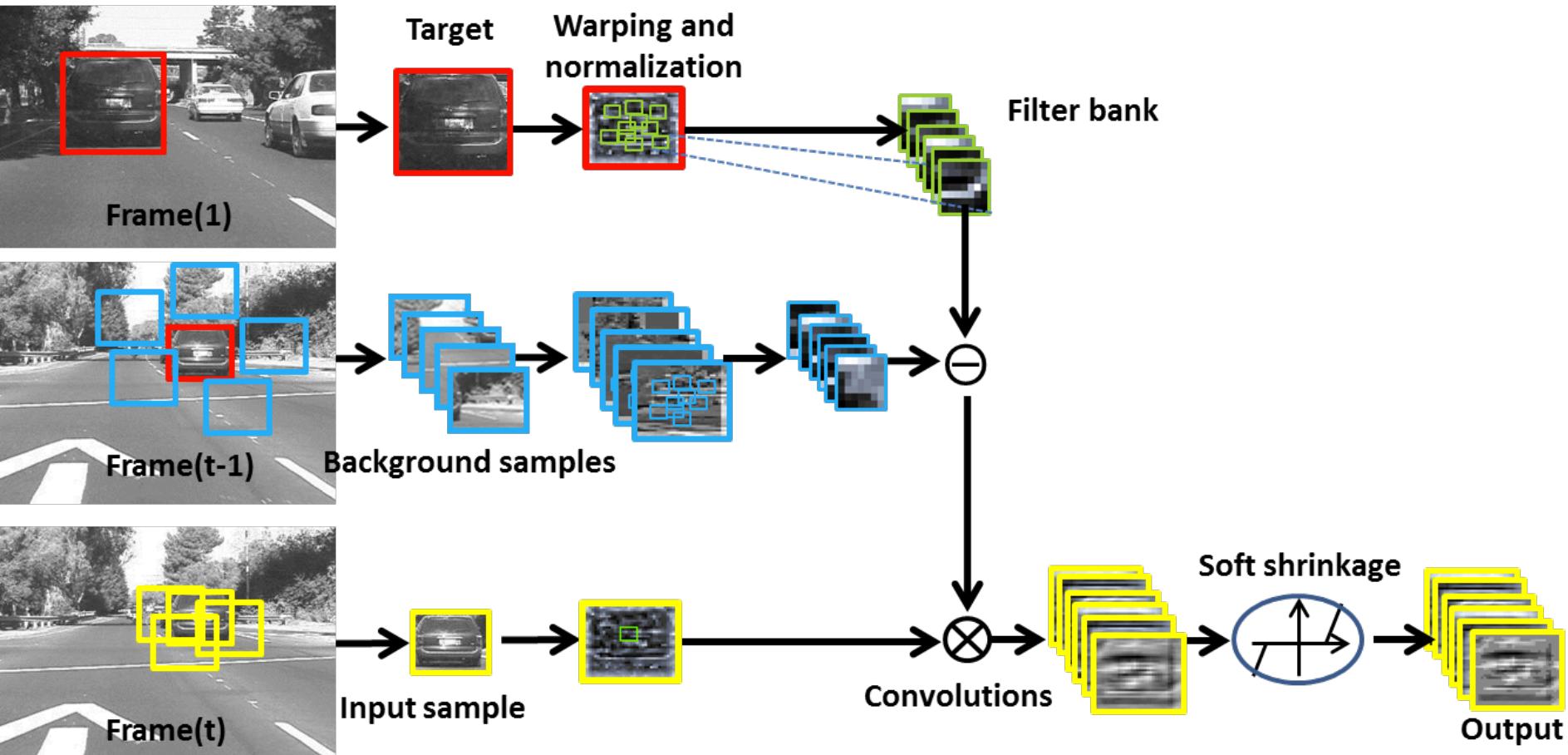
Ejemplo



Ejemplo futbol de humanoides



Estimación de bloques de movimiento



Flujo Óptico

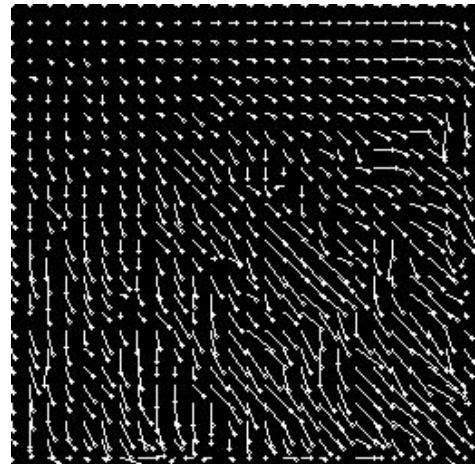
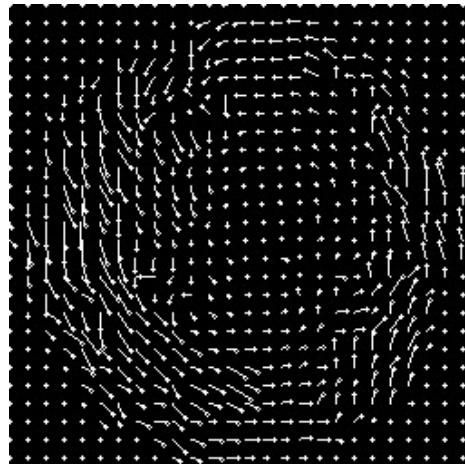
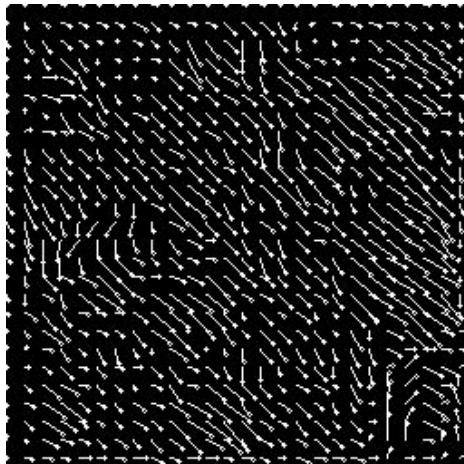
Translation



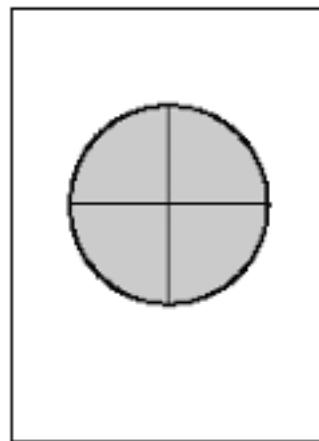
Rotation



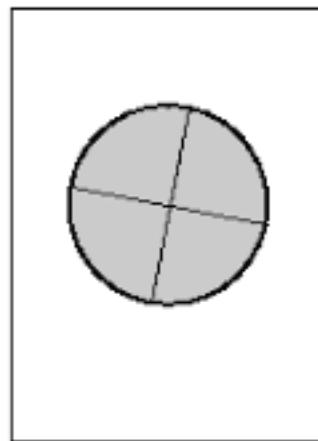
Scaling



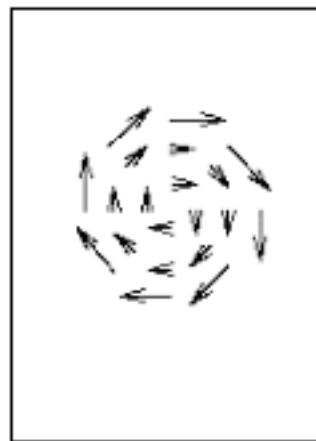
Flujo Óptico



(a)



(b)



(c)

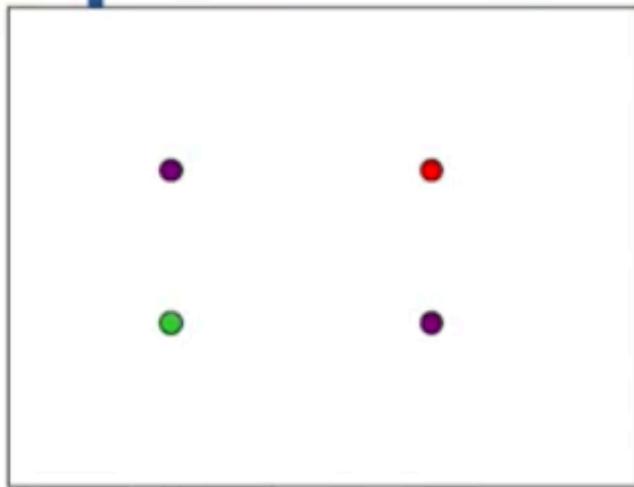
Figure 14.6 *Optical flow: (a) Time t_1 , (b) time t_2 , (c) optical flow.*

Flujo Óptico

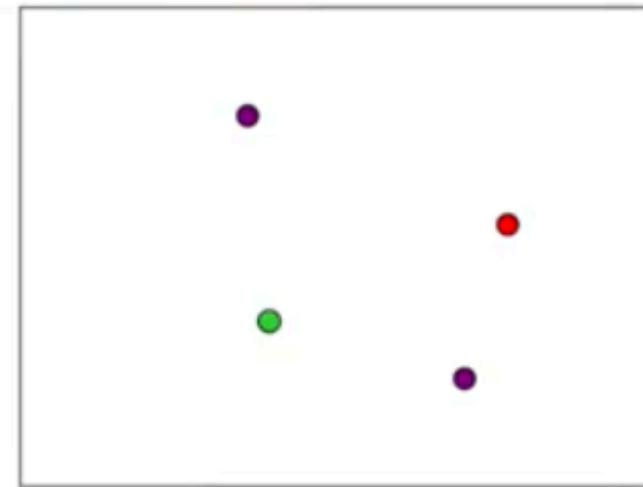
Para cada pixel calcular el vector de velocidad

$$\left(\frac{dx}{dt}, \frac{dy}{dt} \right)$$

Que tan rapido se mueve cada pixel



$H(x, y)$



$I(x, y)$

Para cada pixel calcular el vector de velocidad

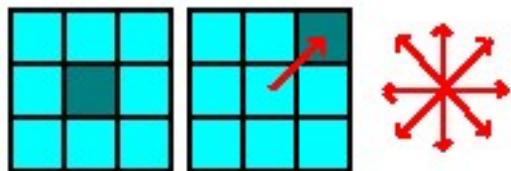
Flujo Óptico

- The optical flow for a pixel (i, j) given 2 successive images k and $k + 1$:

$$\overrightarrow{m_k(i, j)} = (x, y) \quad \text{so that}$$

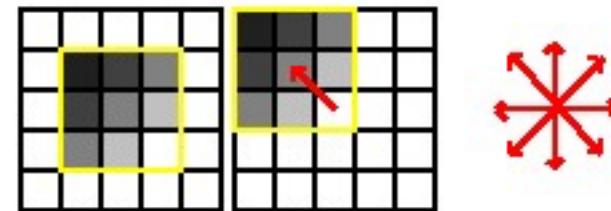
$$|I_k(i, j) - I_{k+1}(i + x, j + y)| \quad (1)$$

is minimum $-1 \leq x \leq 1, -1 \leq y \leq 1$
for



k $k+1$

- More precision: consider a 3×3 window around the pixel:



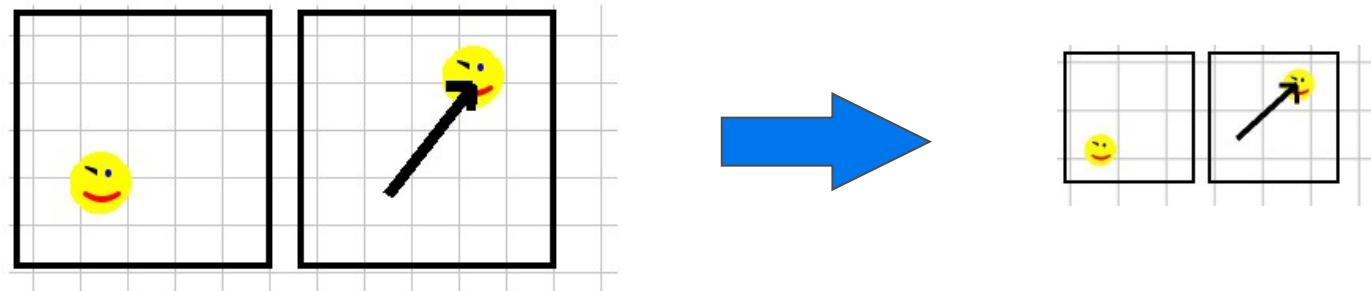
- Optical flow for pixel (i, j) becomes:

$$\overline{m_k(i, j)} = (x, y) \quad \text{so that}$$

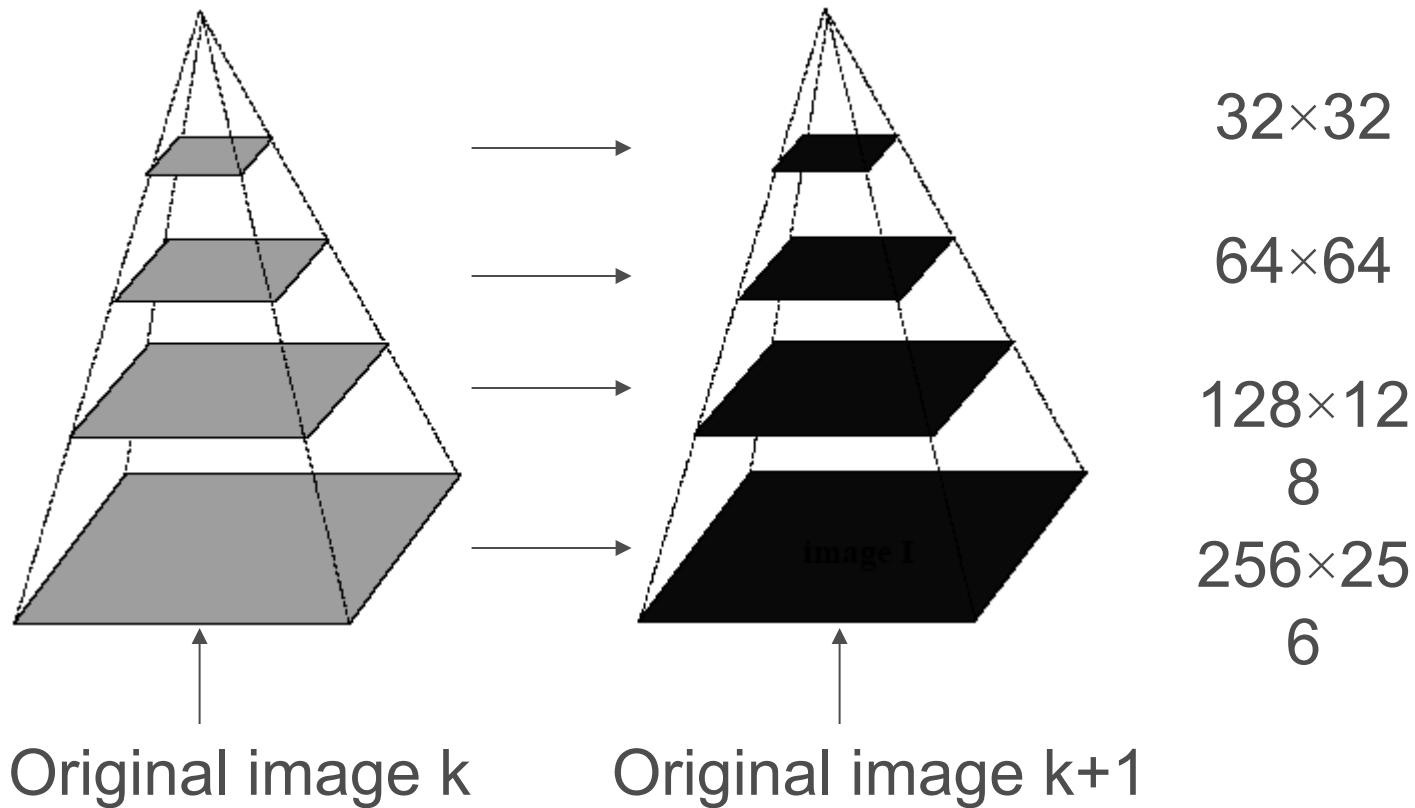
$$\left| \sum_{u=-1}^1 \sum_{v=-1}^1 I_k(i+u, j+v) - \sum_{u=-1}^1 \sum_{v=-1}^1 I_{k+1}(i+u+x, j+v+y) \right| \quad (2)$$

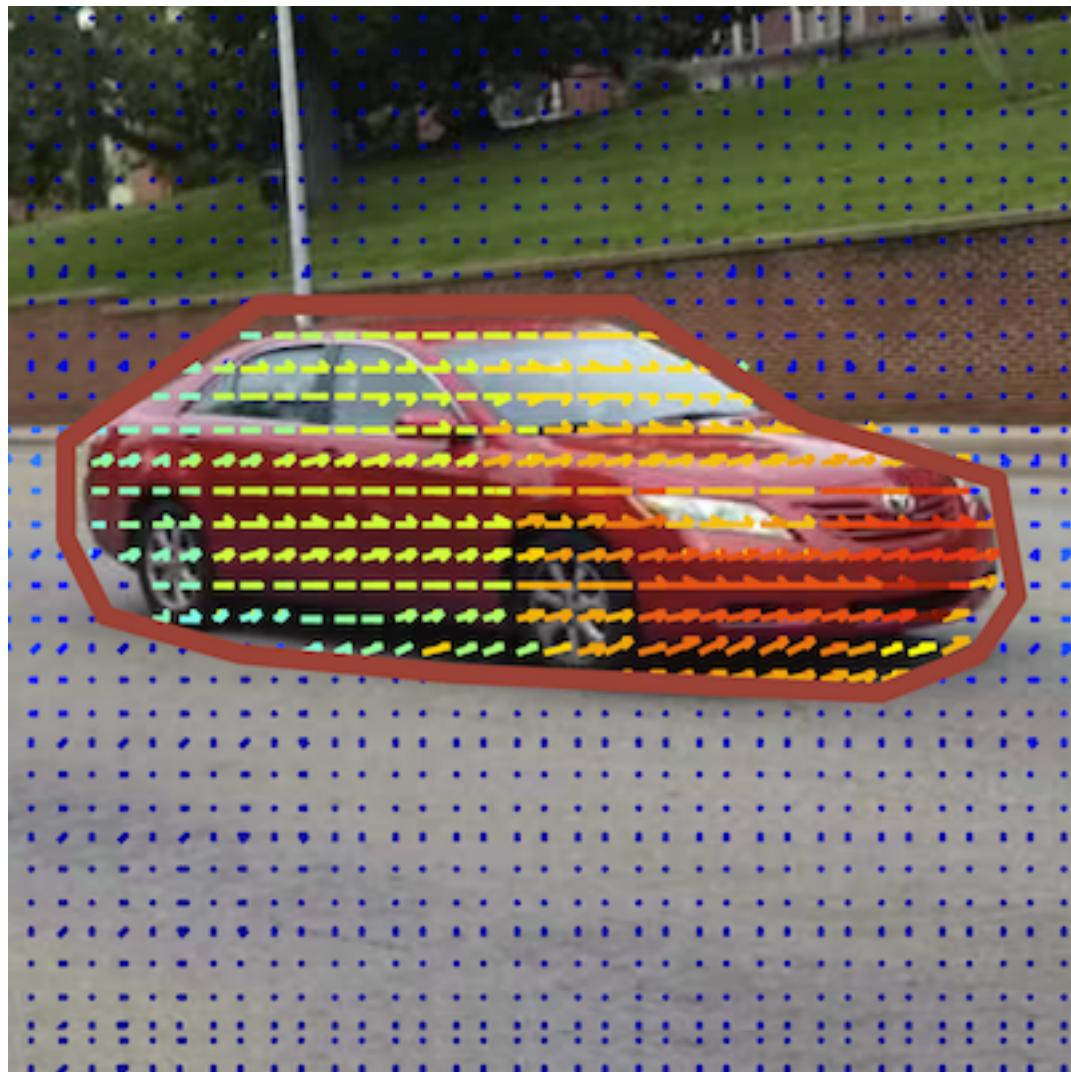
is minimum for $-1 \leq x \leq 1, -1 \leq y \leq 1$

- Reduce the size of the image
=> reduced size of the movement



- Solution: multi-resolution analysis of the images
- Advantages: computing efficiency, stability





- Filtro de Kalman

Filtro de Kalman

Propuesto por Rudolph E. Kalman en 1960

A New Approach to Linear Filtering and Prediction Problems¹

R. E. KALMAN

Research Institute for Advanced Study,²
Baltimore, Md.

The classical filtering and prediction problem is re-examined using the Bode-Shannon representation of random processes and the "state transition" method of analysis of dynamic systems. New results are:

(1) The formulation and methods of solution of the problem apply without modification to stationary and nonstationary statistics and to growing-memory and infinite-memory filters.

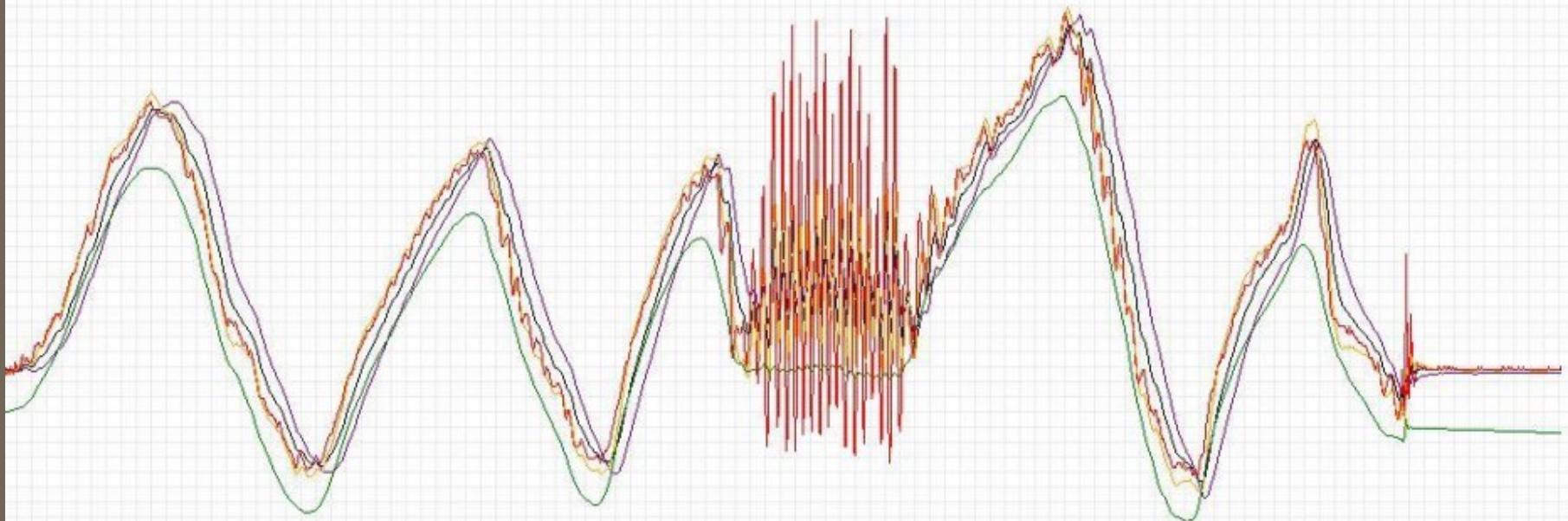
(2) A nonlinear difference (or differential) equation is derived for the covariance matrix of the optimal estimation error. From the solution of this equation the coefficients of the difference (or differential) equation of the optimal linear filter are obtained without further calculations.

(3) The filtering problem is shown to be the dual of the noise-free regulator problem.

The new method developed here is applied to two well-known problems, confirming and extending earlier results.

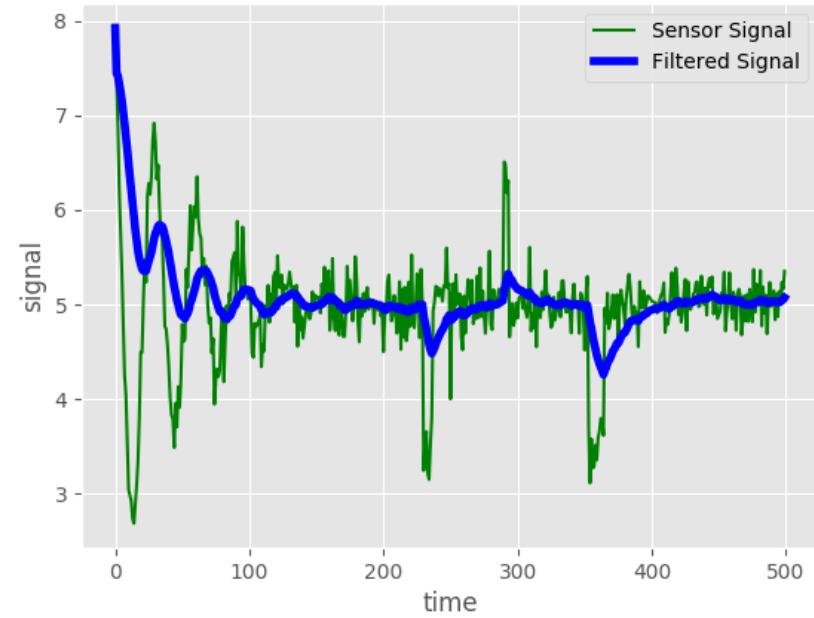
Filtro de Kalman

- Control de sistemas
- Eliminación de ruido en señales
- Seguimiento en visión computacional
- Economía

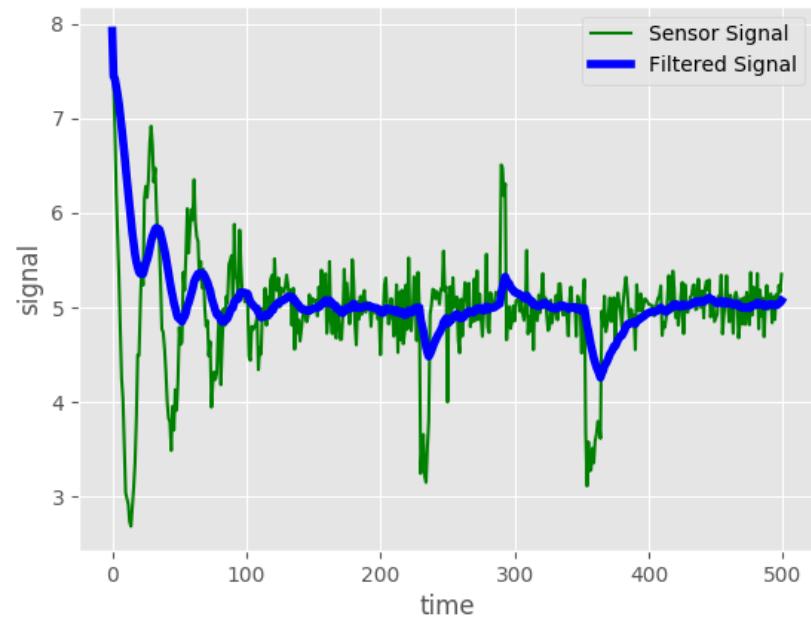
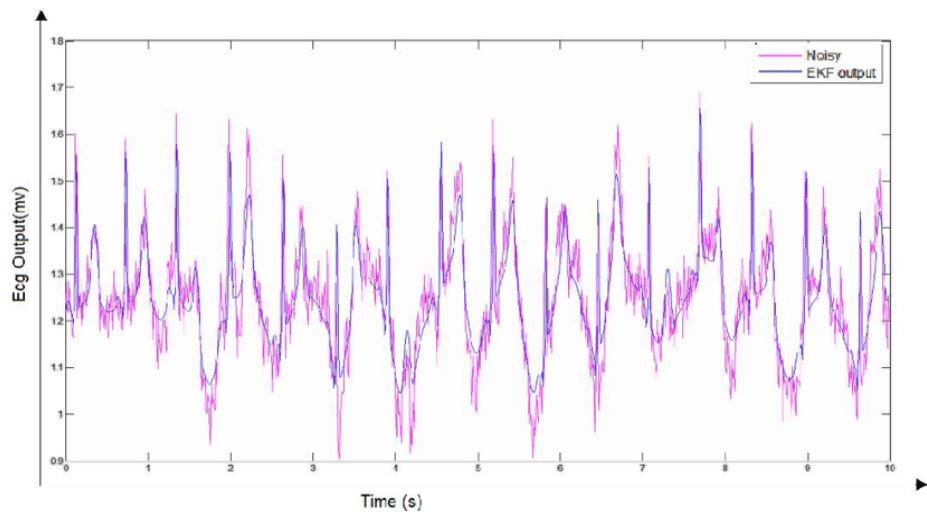


Filtro de Kalman (ej.)

Filtro de Kalman



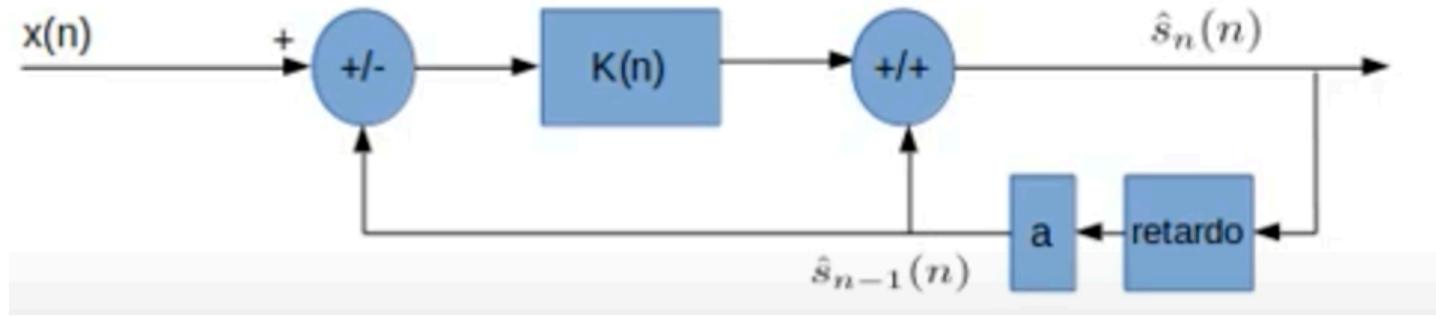
Filtro de Kalman



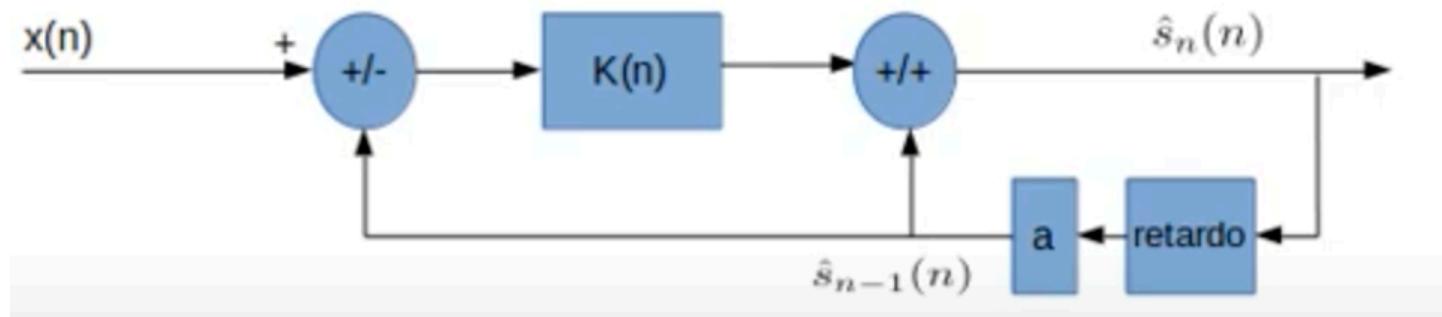
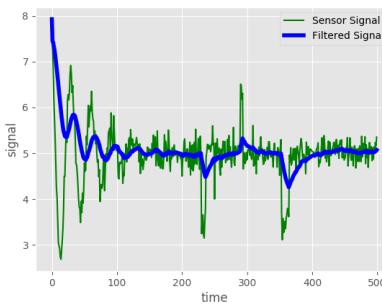
Características

- Espacio de estados
- Adaptable
- Señales estocástica
- Predicción de un instante después mediante los ejemplos anteriores
- Minimiza la covarianza del error estimado

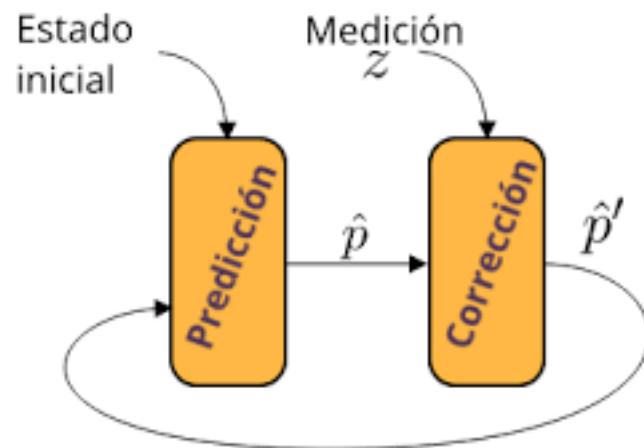
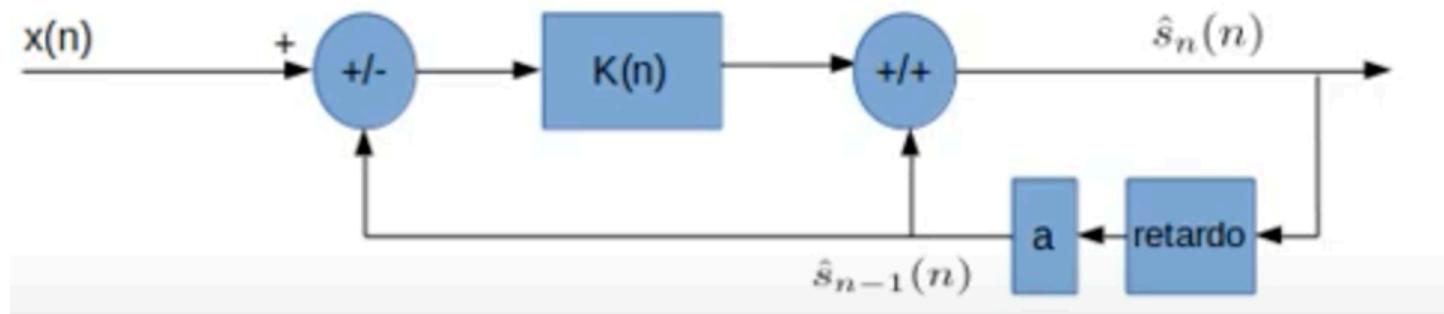
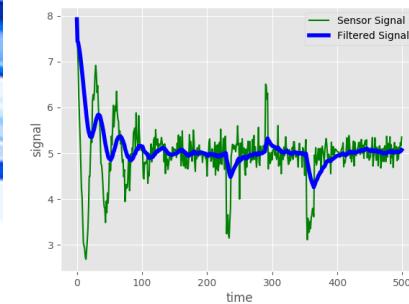
Modelo



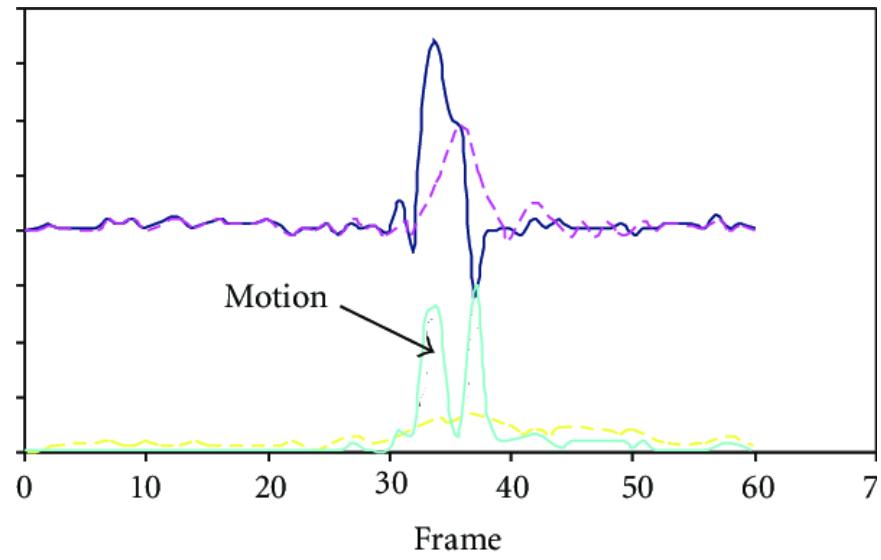
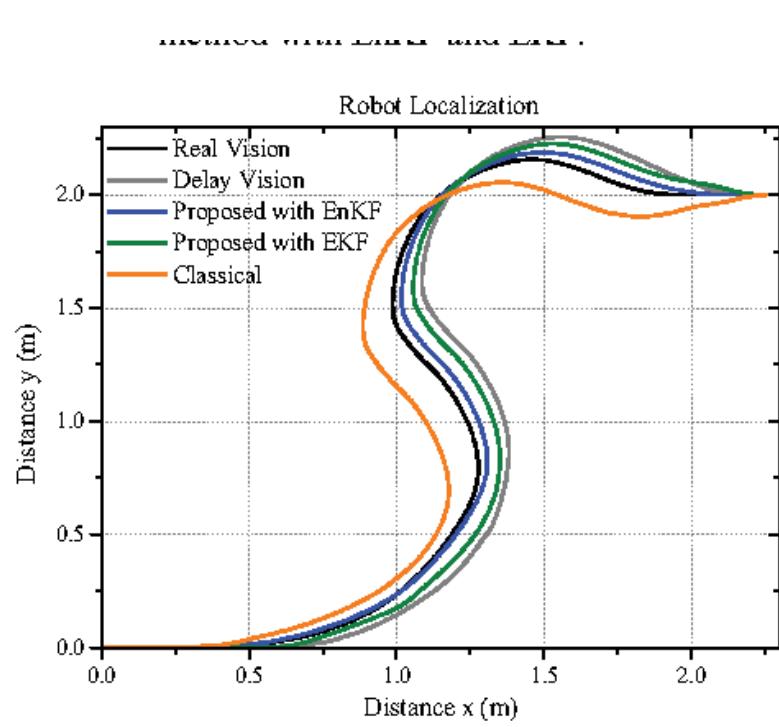
Modelo



Modelo



Estimación de movimiento



the robot localization comparing between the classical meth

- <https://la.mathworks.com/videos/introduction-to-kalman-filters-for-object-tracking-79674>

Ejemplo

Filtro de Particula



Simulación de Montecarlo

- Gracias a casinos de Montecarlo-Mónaco
- Laboratorio Nacional de Los Álamos 2WW
- Idea inicial propuesta por **John von Neumann** y **Stanislaw Ulam**
- Refinada por **Enrico Fermi**, **Nicholas Metropolis**

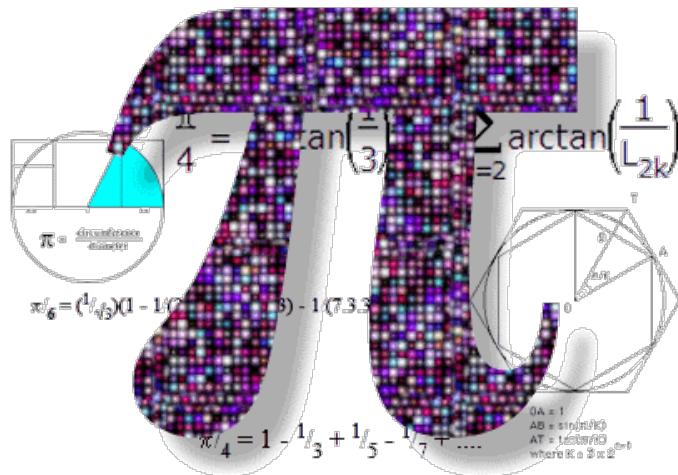
Simulación de Montecarlo

- Puede resolver problemas estocásticos o deterministas
- El error decrece en relación a:

$$\frac{1}{\sqrt{N}}$$

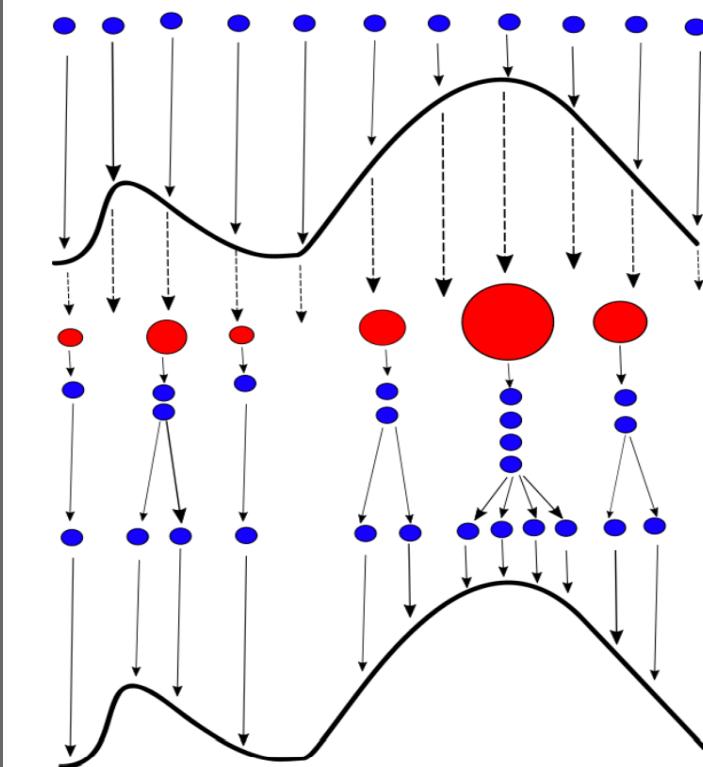
Ejemplo

- Cálculo de la constante PI



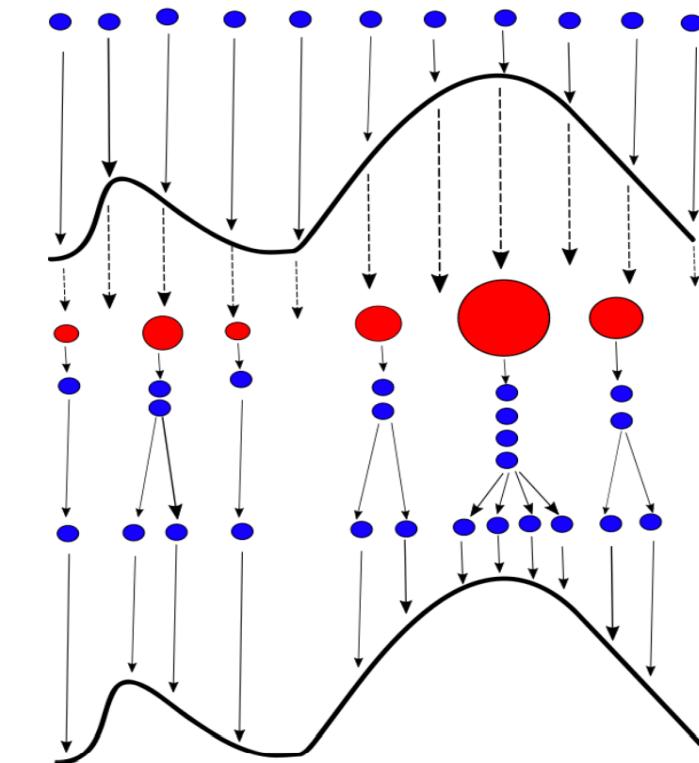
<https://www.youtube.com/watch?v=WJjDr67frtM>

Filtro de partícula



1. Inicializar partículas
2. Asignar pesos a partículas
3. Remuestreo
4. Propagación / predicción

Filtro de partícula



1. Inicializar partículas

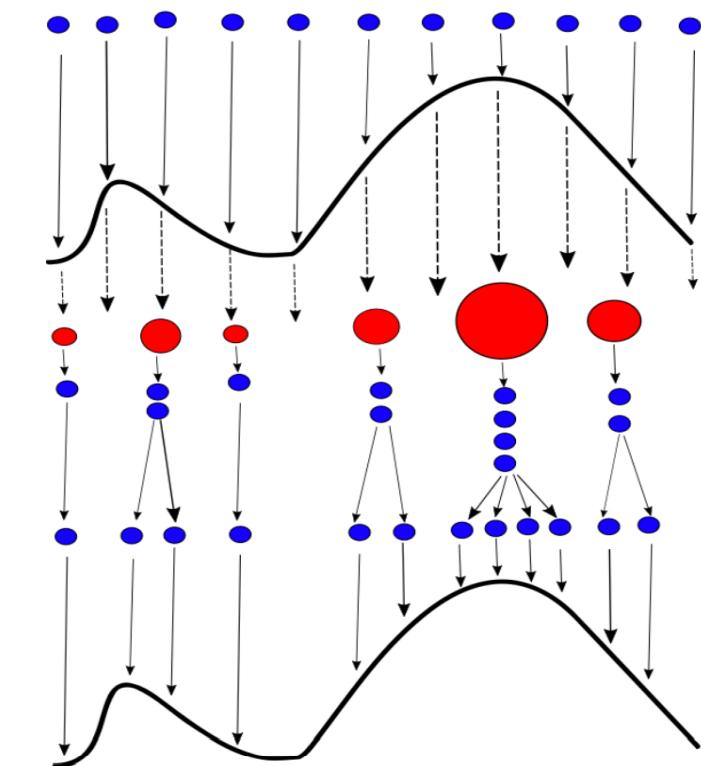
$$x_0^{(i)} \sim P(x_0)$$

2. Asignar pesos a partículas

3. Remuestreo

4. Propagación / predicción

Filtro de partícula



1. Inicializar partículas

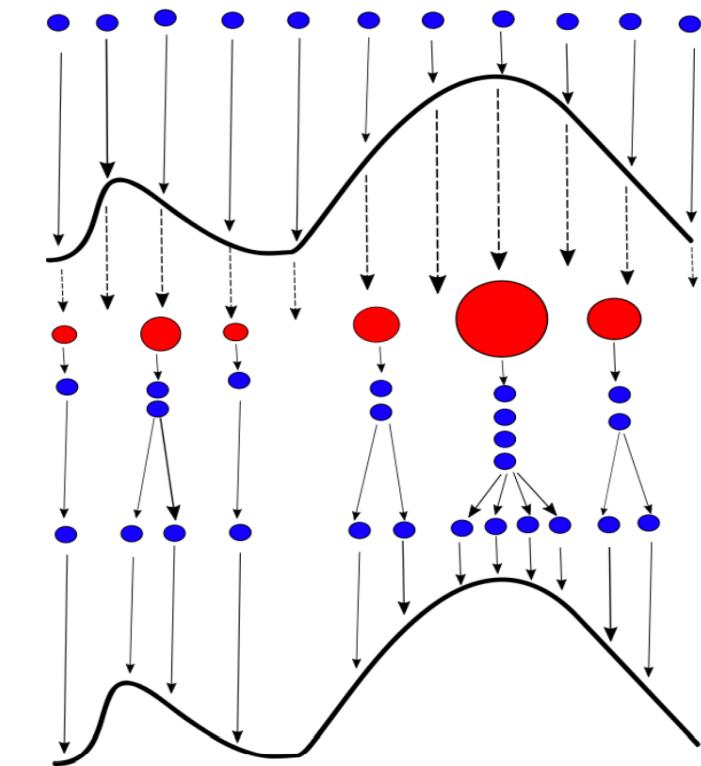
$$x_0^{(i)} \sim P(x_0) \quad w_0^{(i)} = \frac{1}{N}$$

2. Asignar pesos a partículas

3. Remuestreo

4. Propagación / predicción

Filtro de partícula



1. Inicializar partículas

$$x_0^{(i)} \sim P(x_0) \quad w_0^{(i)} = \frac{1}{N}$$

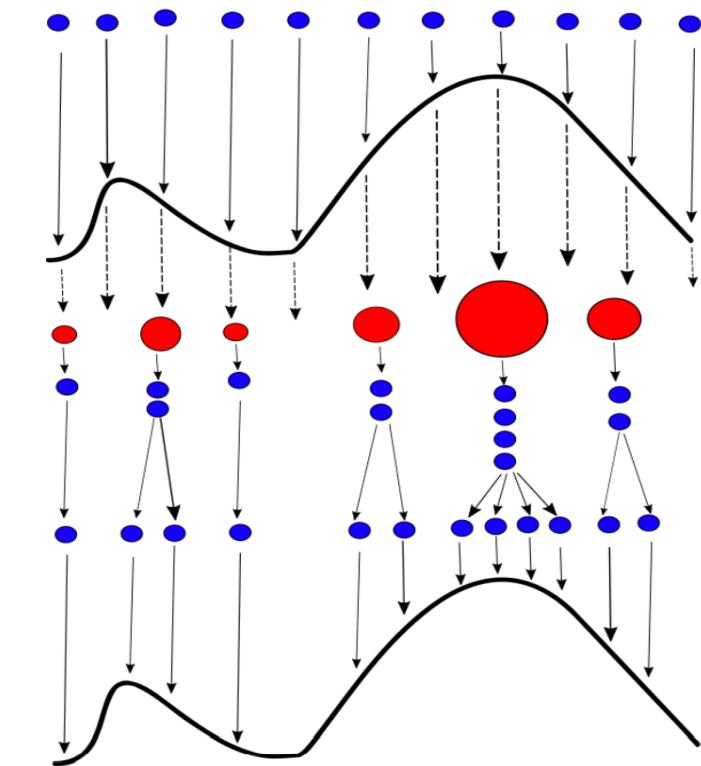
$$\tilde{w}_k^{(i)} = \frac{w(x^{(i)})}{\sum_{j=1}^N w(x^{(j)})}$$

2. Asignar pesos a partículas

3. Remuestreo

4. Propagación / predicción

Filtro de partícula



1. Inicializar partículas

$$x_0^{(i)} \sim P(x_0) \quad w_0^{(i)} = \frac{1}{N}$$

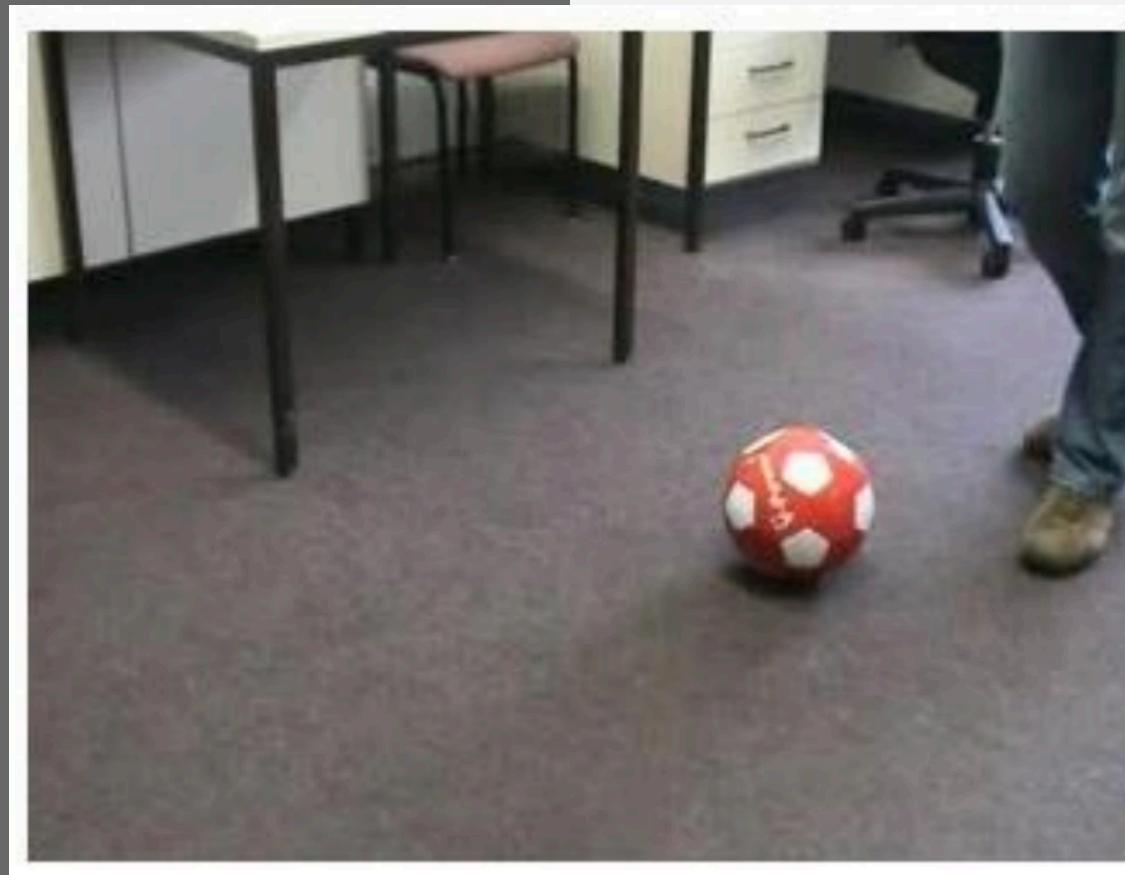
2. Asignar pesos a partículas

3. Remuestreo

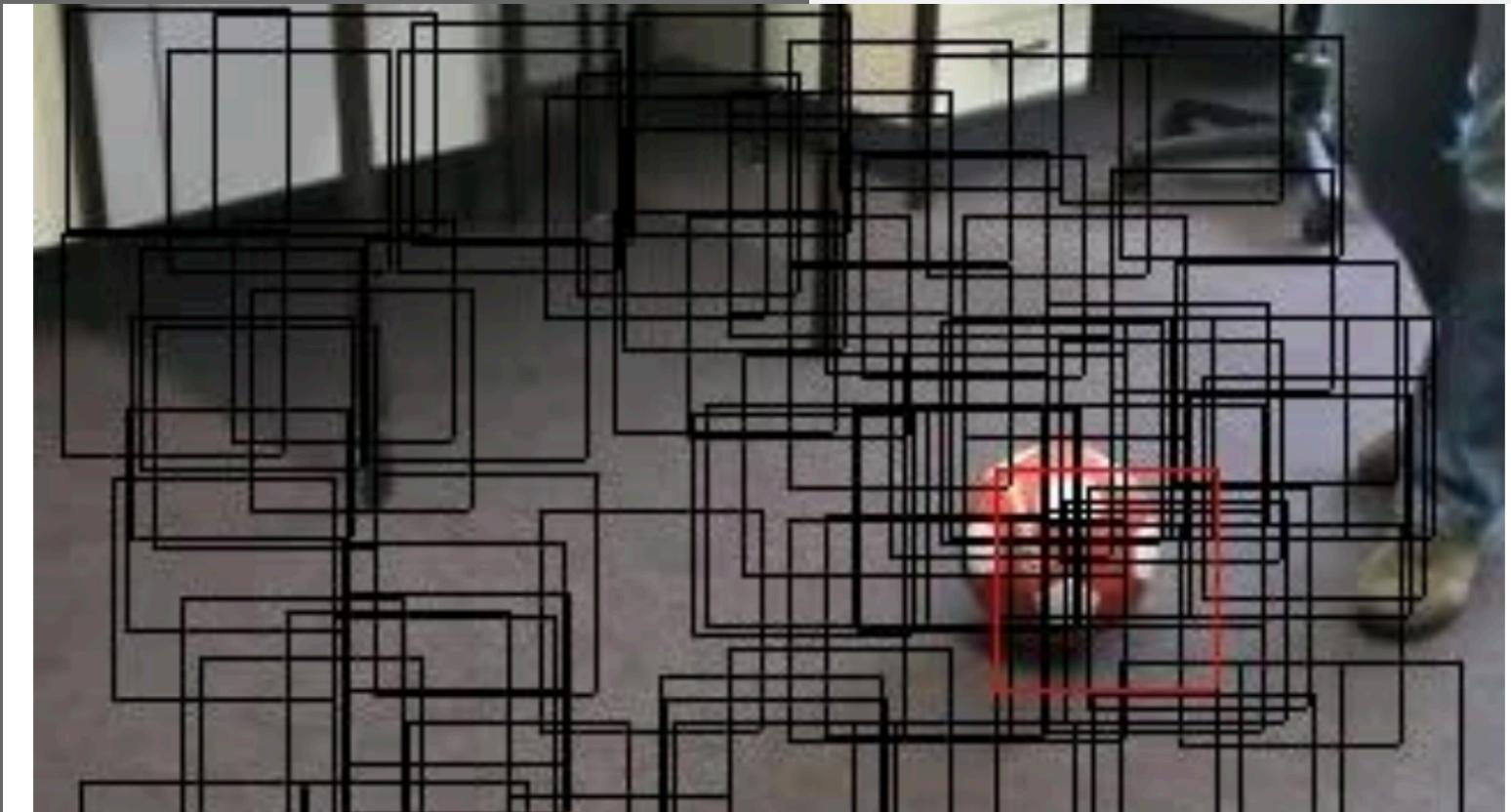
$$\tilde{w}_k^{(i)} = \frac{w(x^{(i)})}{\sum_{j=1}^N w(x^{(j)})}$$

4. Propagación / predicción

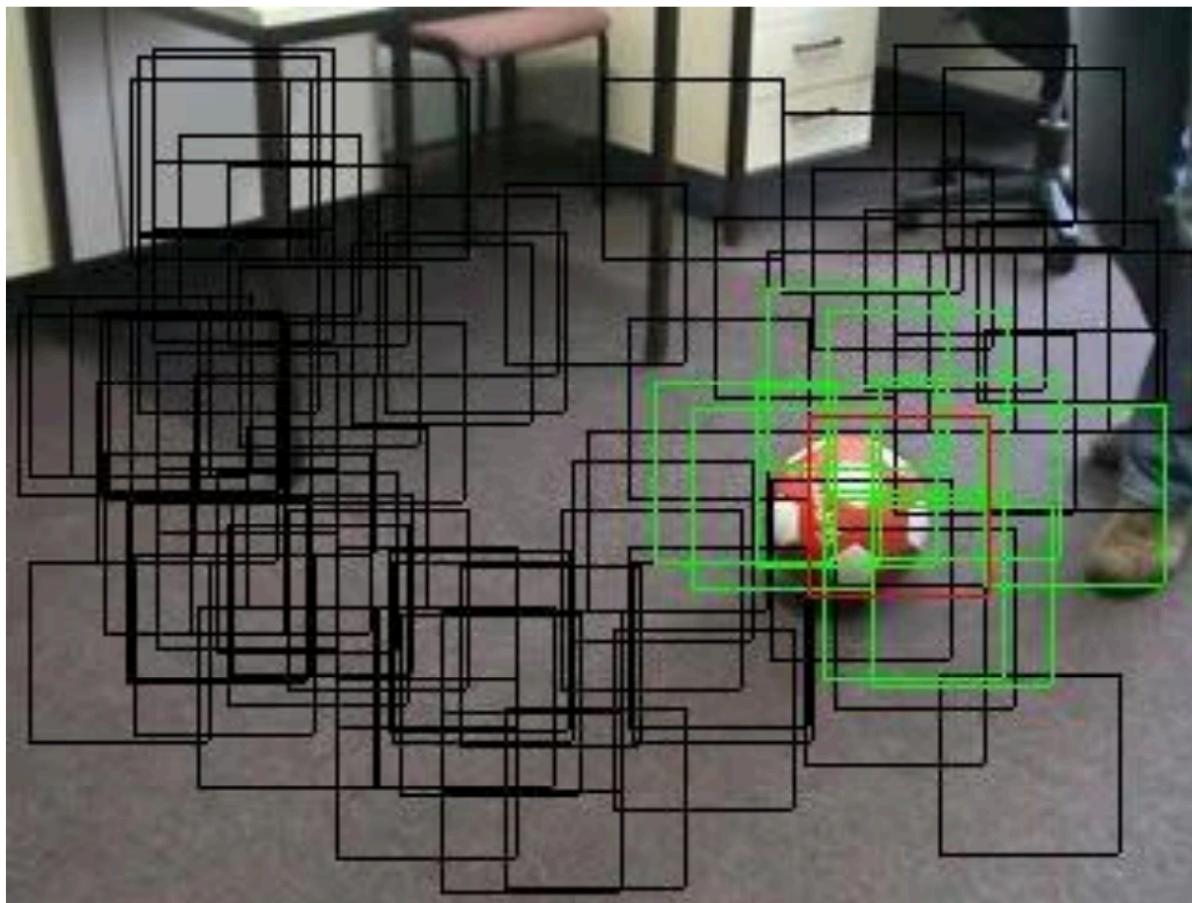
Filtro de partícula



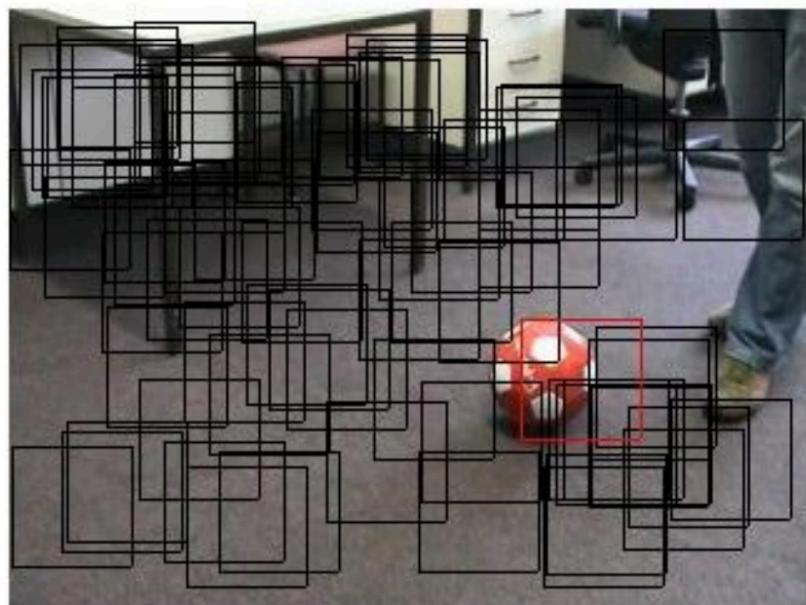
1) Inicialización



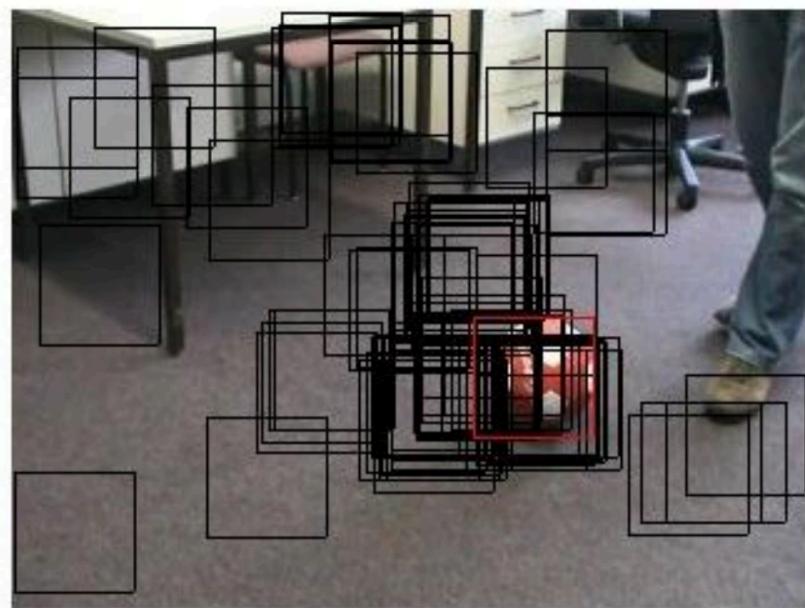
2) Asignación de pesos



3) Re-muestreo de partículas



Cuadro 32



Cuadro 33

A photograph of a red and white soccer ball resting on a grey carpeted floor. In the background, there's a dark wooden bench and a white wall with some posters or notices pinned to it. A small blue rectangular graphic is located in the top right corner of the slide.

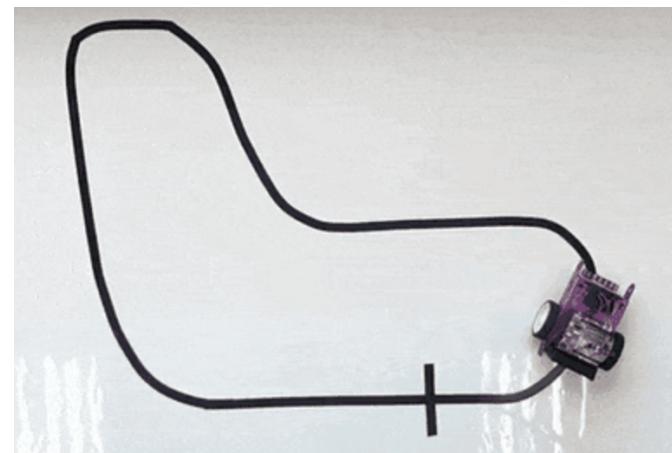
4) Predicción

Demo

- <https://www.youtube.com/watch?v=-EZjTNIGNJs>

Tarea

- Hacer seguimiento usando:
- Flujo Óptico
- Filtro de Kalman
- Filtro de Partícula



Proyecto Final (para el lunes 2 parcial)



- Presentación
 - Estado del arte
 - Objetivos
 - Pre-procesamiento de imágenes (es necesario ya contar con la BD)

Proyecto Final



- Escrito formato IEEE-EMBC
(<https://embc.embs.org/2018/author-instructions-for-contributed-papers-full/>)
- (Título, resumen, introducción, metodología, resultados, discusión y conclusiones. ver formato)
- Presentación oral